

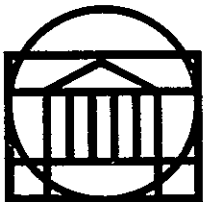
(NASA-CR-149488) AN INVESTIGATION OF
POTENTIAL APPLICATIONS OF OP-SAPS:
OPERATIONAL SAMPLED ANALOG PROCESSORS Final
Report (Virginia Univ.) 143 P HC A07/MF A01

N77-16806

Unclas

CSCI 09B G3/63 13256

RESEARCH LABORATORIES FOR THE ENGINEERING SCIENCES



SCHOOL OF ENGINEERING AND APPLIED SCIENCE

University of Virginia

Charlottesville, Virginia 22901

A Report

AN INVESTIGATION OF POTENTIAL APPLICATIONS OF
OP-SAPS: OPERATIONAL SAMPLED ANALOG PROCESSORS

Final Report

Grant No. NSG-1223

Submitted to:

NASA Scientific & Technical Information Facility
P. O. Box 8757
Baltimore/Washington International Airport
Maryland 21240

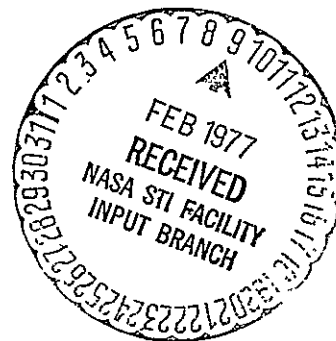
Submitted by:

E. A. Parrish
Associate Professor

E. S. McVey
Professor

Report No. JVA/528011/EE77/103

February 1977



RESEARCH LABORATORIES FOR THE ENGINEERING SCIENCES

Members of the faculty who teach at the undergraduate and graduate levels and a number of professional engineers and scientists whose primary activity is research generate and conduct the investigations that make up the school's research program. The School of Engineering and Applied Science of the University of Virginia believes that research goes hand in hand with teaching. Early in the development of its graduate training program, the School recognized that men and women engaged in research should be as free as possible of the administrative duties involved in sponsored research. In 1959, therefore, the Research Laboratories for the Engineering Sciences (RLES) was established and assigned the administrative responsibility for such research within the School.

The director of RLES—himself a faculty member and researcher—maintains familiarity with the support requirements of the research under way. He is aided by an Academic Advisory Committee made up of a faculty representative from each academic department of the School. This Committee serves to inform RLES of the needs and perspectives of the research program.

In addition to administrative support, RLES is charged with providing certain technical assistance. Because it is not practical for each department to become self-sufficient in all phases of the supporting technology essential to present-day research, RLES makes services available through the following support groups: Machine Shop, Instrumentation, Facilities Services, Publications (including photographic facilities), and Computer Terminal Maintenance.

A Report

AN INVESTIGATION OF POTENTIAL APPLICATIONS OF
OP-SAPS: OPERATIONAL SAMPLED ANALOG PROCESSORS

Final Report

Grant No. NSG-1223

Submitted to:

NASA Scientific & Technical Information Facility
P. O. Box 8757
Baltimore/Washington International Airport
Maryland 21240

Submitted by:

E. A. Parrish
Associate Professor

E. S. McVey
Professor

Department of Electrical Engineering
RESEARCH LABORATORIES FOR THE ENGINEERING SCIENCES
SCHOOL OF ENGINEERING AND APPLIED SCIENCE
UNIVERSITY OF VIRGINIA
CHARLOTTESVILLE, VIRGINIA

Report No. UVA/528011/EE77/103
February 1977

Copy No. 1

I. INTRODUCTION

This report presents results achieved during the July 1, 1975, through October 31, 1976, period on the application of OP-SAPs (operational sampled analog processors) in pattern recognition systems. The investigations included four areas: (1) human face recognition; (2) a high-speed programmable transversal filter system; (3) discrete word (speech) recognition; and (4) a resolution enhancement system.

Detailed information is available in published journal articles (see Appendix), masters theses, and a doctoral dissertation to be completed shortly. Thus, this report summarizes those research projects that are documented elsewhere. In particular, the following publications are based on this research:

1. E. A. Parrish, Jr., and E. S. McVey, "Implications of Charge-Coupled Devices for Pattern Recognition," IEEE Trans. on Comp., pp. 1146-52, Nov. 1976.
2. E. S. McVey and E. A. Parrish, Jr., "Charge-Coupled Devices Applied to Control and Pattern Recognition Systems," Mundo Electronico, Dec. 1976.
3. E. S. McVey and E. A. Parrish, Jr., "The Application of Charge-Coupled Device Processors in Automatic Control Systems," submitted to IEEE Trans. on Auto. Cont.
4. E. W. Evers, "Application of CCDs to Fourier Transform Weighted Transversal Filters," M.S. Thesis, University of Virginia, Sept. 1976.
5. C. S. Dreher, "Human Face Identification Using CCD Discrete Fourier Transform Filters," M.S. Thesis in preparation, University of Virginia, Jan. 1977.
6. R. Scher, "A Programmable Discrete Analog Processor for Pattern Recognition," Ph.D. Dissertation in preparation, University of Virginia, May 1977.

The last two publications will contain additional results obtained after expiration of the present grant.

II. HUMAN FACE IDENTIFICATION

Introduction

Of particular interest in pattern recognition studies is the discrete Fourier transform and an analogous transform, the Chirp [5] z-transform [1], [2], [16]. Implementation of these transforms has, up until recently, been restricted to digital computer simulation via the FFT (too slow for real-time processing) or dedicated array processors (very expensive) [3]. Charge-coupled devices promise to circumvent both these difficulties [17] by offering real-time array processing at chip-set prices.

The theoretical use of transversal filters and transforms has been discussed in the literature [4], [6], [14]. A working charge-coupled Chirp z-transform device has also been fabricated and tested [7]. While this device is a laboratory model only, the path is clearing for commercial manufacture of CCD building blocks which perform standard transform and filter operations [8].

The pattern recognition system described herein employs the discrete Fourier transform (DFT). The brief description which follows is intended as a refresher on the DFT and should help to make the included graphs easier to interpret [9], [10], [11].

1. The DFT is a sampled data system and, as such, imposes a finite, sampled record-length restriction on the data. The effect of this is to presuppose periodic data (Fig. 2-1a).
2. Because of the implied periodicity of the DFT, correlation (and convolution) requires augmenting the data record with zeros to twice the original record size (Fig. 2-1b).
3. The DFT is symmetric on the frequency axis (Fig. 2-1c). For example, for a 64-point transform

$$f_0 = \text{DC term}$$

$$f_1 = \text{1st harmonic} = f_{64}$$

$$f_{31} = f_{33}$$

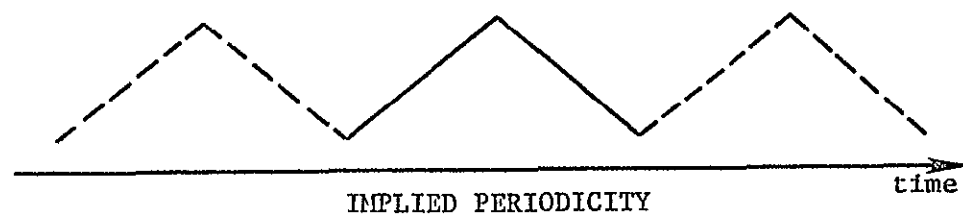
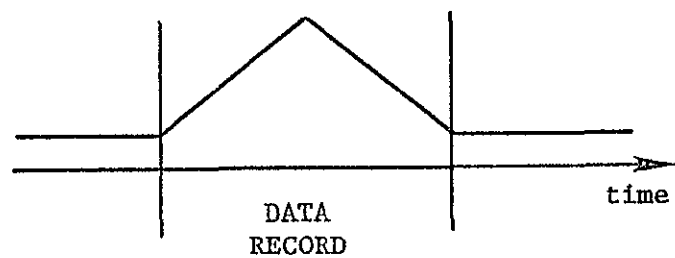


Figure 2-1a. Implied Periodicity of DFT.

7

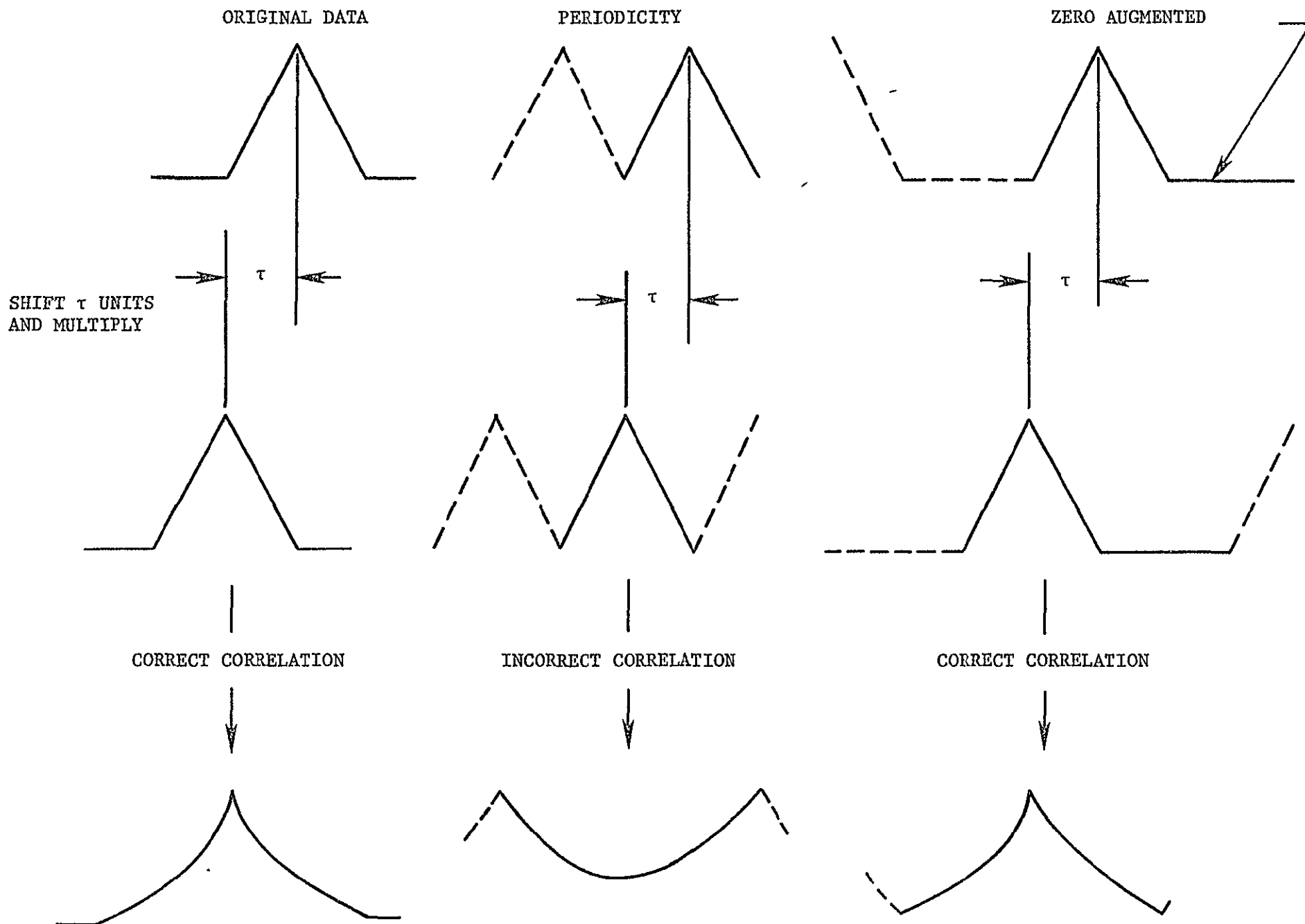


Figure 2-1b. Zero Augmentation for Correct Correlation

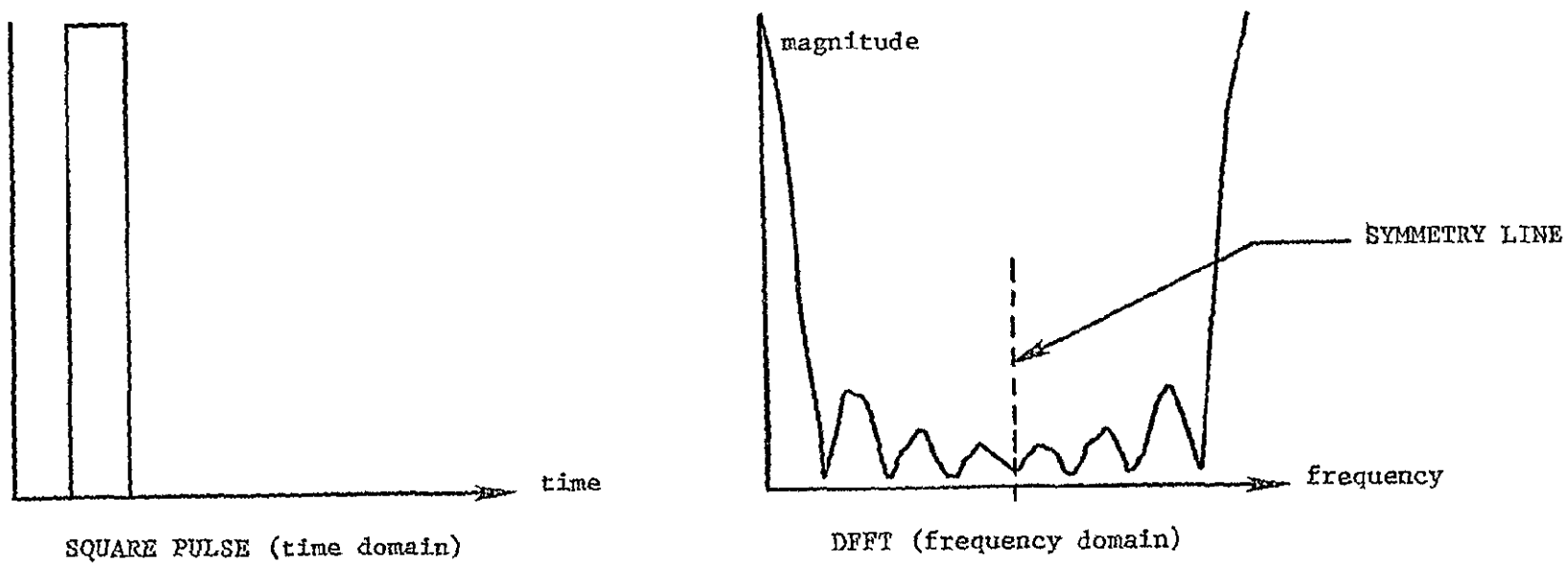


Figure 2-1c. Discrete Fast Fourier Transform Illustrating Frequency Symmetry

Normalized Cross Correlation

The process employed with this pattern recognition system is that of normalized cross correlation. Given a two-dimensional picture, $g(x,y)$, and a template, $t(x,y)$, a template matching scheme can be defined as

$$E(m,n) = (\sum_i \sum_j (g(i,j) - t(i-m,j-n))^2)^{1/2} \quad (2-1)$$

Removing the square root and expanding

$$E^2(m,n) = \sum_i \sum_j g^2(i,j) - 2g(i,j)t(i-m,j-n) + t^2(i-m,j-n) \quad (2-2)$$

Note that, for any template, $t^2(i-m,j-n)$ is constant and, hence, can be subtracted from the index $E^2(m,n)$ with no loss of information. The term $g^2(i,j)$ is defined as the picture energy. If the picture energy varies only slightly over each scene, it, too, can be subtracted. Defining the resultant expression as the cross correlation between scene $g(x,y)$ and template $t(x,y)$, we arrive at

$$R_{gt}(m,n) = \sum_i \sum_j g(i,j)t(i-m,j-n) \quad (2-3)$$

and the normalized cross correlation as

$$N_{gt}(m,n) = R_{gt}(m,n) / (\sum_i \sum_j g(i,j)^2)^{1/2} \quad (2-4)$$

The normalized cross correlation coefficient is then defined as

$$\max_{m,n} (N_{gt}(m,n)) = N_{g_k t}(T_x, T_y) \quad (2-5)$$

This may also be computed via transform techniques as

$$N_{gt}(m,n) = F^{-1}(G(f_x, f_y) \hat{T}(f_x, f_y)) / (\sum_i \sum_j g(i,j)^2)^{1/2} \quad (2-6)$$

where F^{-1} implies the inverse Fourier transform, $G(f_x, f_y)$ is the Fourier

transform of $g(x,y)$, and $\hat{T}(f_x, f_y) = F(t(-x, -y))$, i.e., the data is flipped about a diagonal axis prior to being transformed [12].

Generalized Recognition System Using $N_{g_k t}(T_x, T_y)$

Given a set of samples $\{g_1(x,y) \cdots g_N(x,y)\}$, a normalized cross correlation recognition system could be structured as in Fig. 2-2. Before a system such as this could be configured, it would be necessary to answer the following questions:

1. How much data is necessary to establish a good template match?
(i.e., what is the two-dimensional information bandwidth?)
2. Is the criterion satisfied that $\sum_i \sum_j g(i,j)$ be approximately constant over all scenes?
3. What degree of accuracy can be expected?

Bandwidth

To investigate question 1 it was decided to first obtain as much data as possible in order to observe the widest possible information bandwidth. A data acquisition system which sampled facial scenes taken from a closed circuit TV camera was employed. The sampling frequency was set at 203 KHz, which resulted in eight samples per video line. A data record of 230 lines was established, giving a total of 1,840 data points. The data was quantized via a 10-bit A/D converter. To satisfy the Nyquist sampling criterion the video was prefiltered to 100 KHz bandwidth.

A two-dimensional discrete Fast Fourier Transform (DFFT) of a typical facial scene is shown in Fig. 2-3. (Note the symmetry along the horizontal and vertical lines, as mentioned previously. The horizontal axis contains eight spatial frequency components, while the vertical contains 256. Fold-over occurs at the fifth and 129th components, respectively.) As is evident, the power spectrum in the vertical dimension levels off after approximately 32 spatial harmonics. (Three dB down occurs after the third harmonic.) Thus, we can conclude that facial scenes are essentially composed of low spatial frequencies.

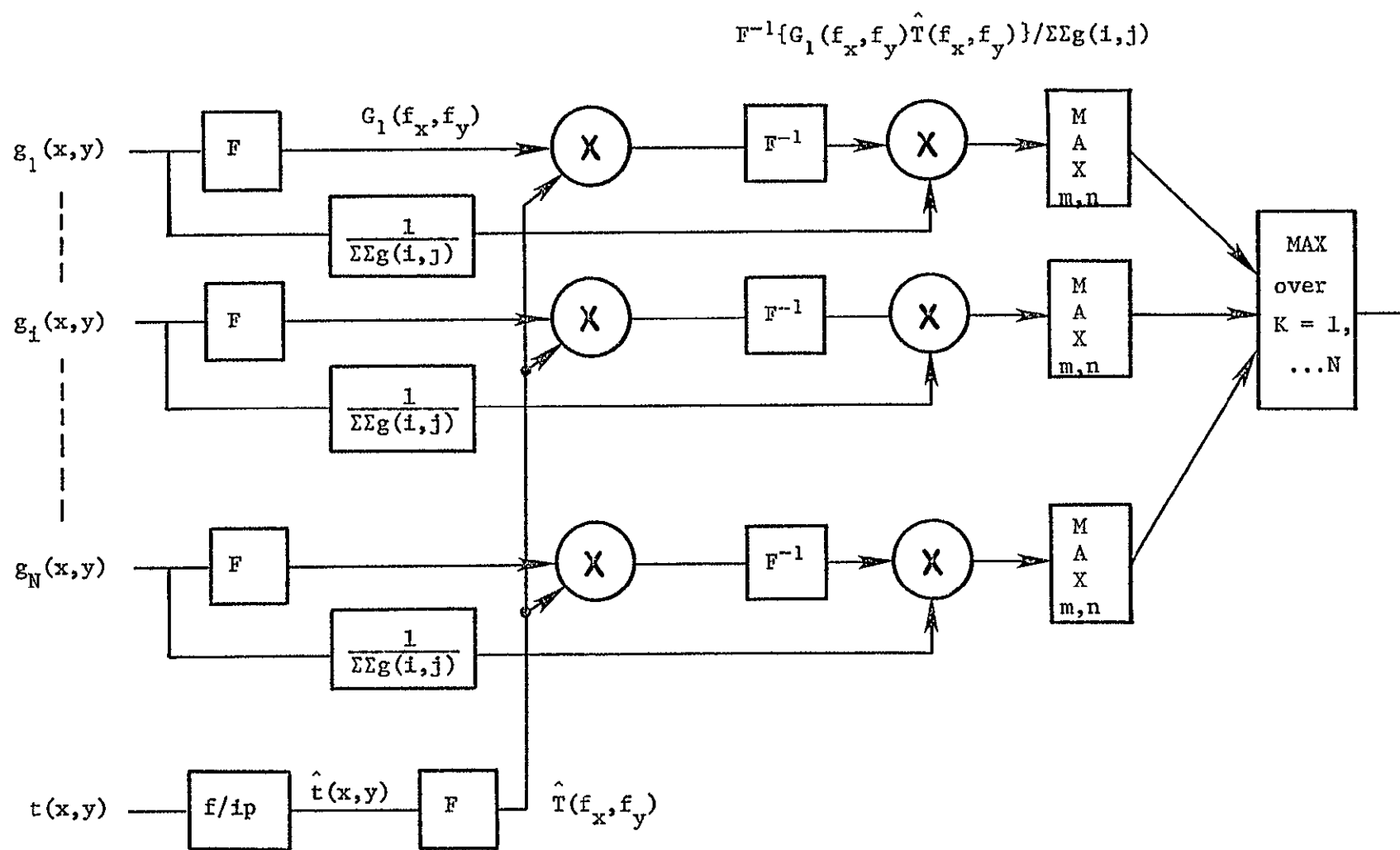
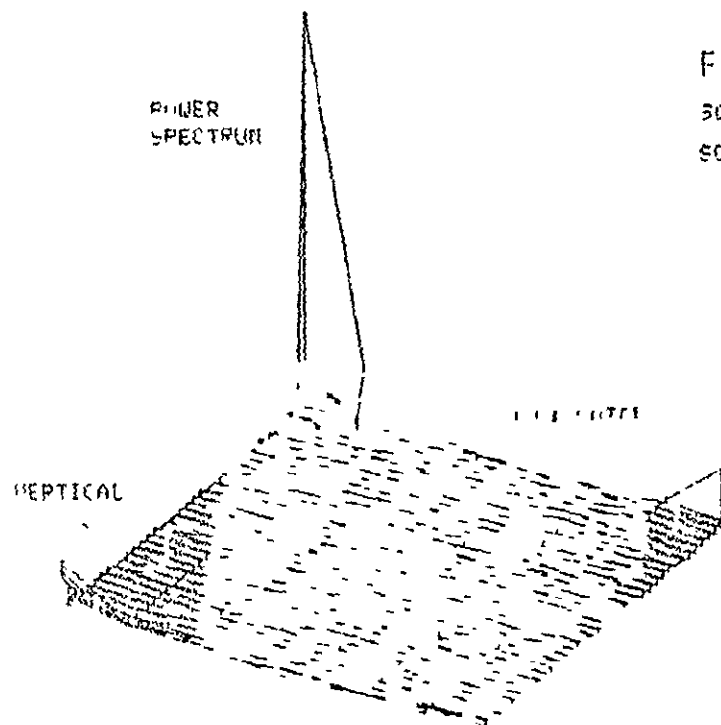


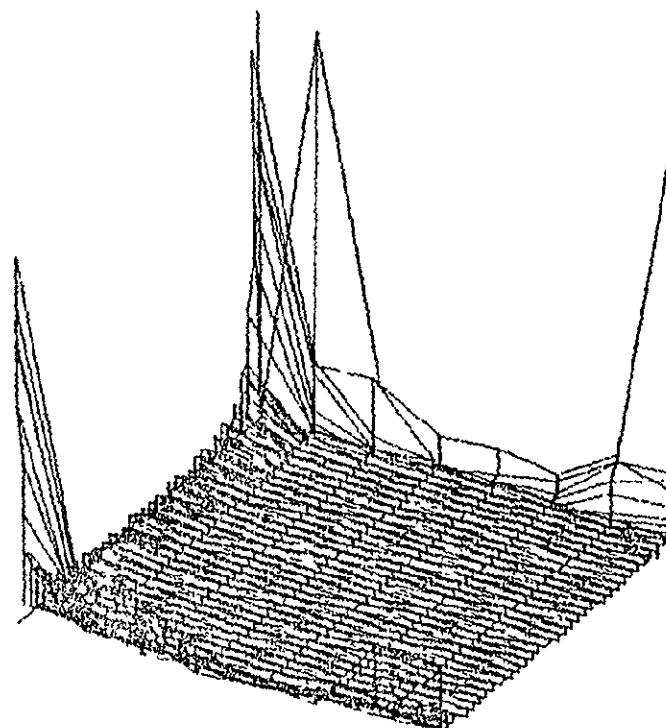
Figure 2. A Normalized Cross Correlation Recognition System



FACIAL POWER SPECTRUM

SCENE DATA: 8x256 HANNING WINDOWED

SCENE: JJS1 (AA1) 10 BITS



C T Drah-r
12-14-76 UUa

DC TERM SUPPRESSED TO EMPHASIZE HIGHER SPATIAL FREQUENCY TERMS.

Figure 2-3. Facial Power Spectrum (8 x 256).

To further investigate the power spectrum the scene was augmented by adding zeros in the horizontal direction, while averaging together every four lines of data in the vertical. (This averaging process can be performed with almost no loss in information, because averaging is akin to low-pass filtering and will thus preserve the low spatial frequencies already determined to contain the most power.) Augmenting the data horizontally increases the frequency resolution in that direction. The result is shown in Fig. 2-4.

Low frequency terms are most apparent in Fig. 2-4. This has decidedly favorable practical aspects, since it implies the sampling rate can be reduced (and likewise, the number of samples necessary to maintain the power bandwidth), while retaining most of the information. Thus, slower and less expensive devices with less memory are required to adequately define a sampled facial scene.

Scene Power

The criterion of question 2 was investigated by finding $(\sum_i \sum_j g(i,j)^2)^{1/2}$ over a data base of 40 facial scenes. The results are given in Table 1.

The average power in each scene (10-bit data) is 15233.74, with a standard deviation of 659.06 or approximately four percent of the total power and thus satisfies Criterion 2.

System Implementation

To judge actual recognition performance a system was simulated. The system flow chart is shown in Fig. 2-5. A detailed description follows:

1. FILE NAMES - All data files are placed on magnetic disk for quick access. Each file is assigned a number and the prefix AA for ease of identification.
2. NUMBER OF BITS - An option is given as to how many bits of the 10-bit (1,024 gray levels) words are to be used. The effect of using fewer bits is to increase the coarseness of the gray level quantization, by right-shifting the data word (10-N) bits, where N = bit specification (Fig. 2-6).

ORIGINAL PAGE IS
OF POOR QUALITY

POWER SPECTRUM

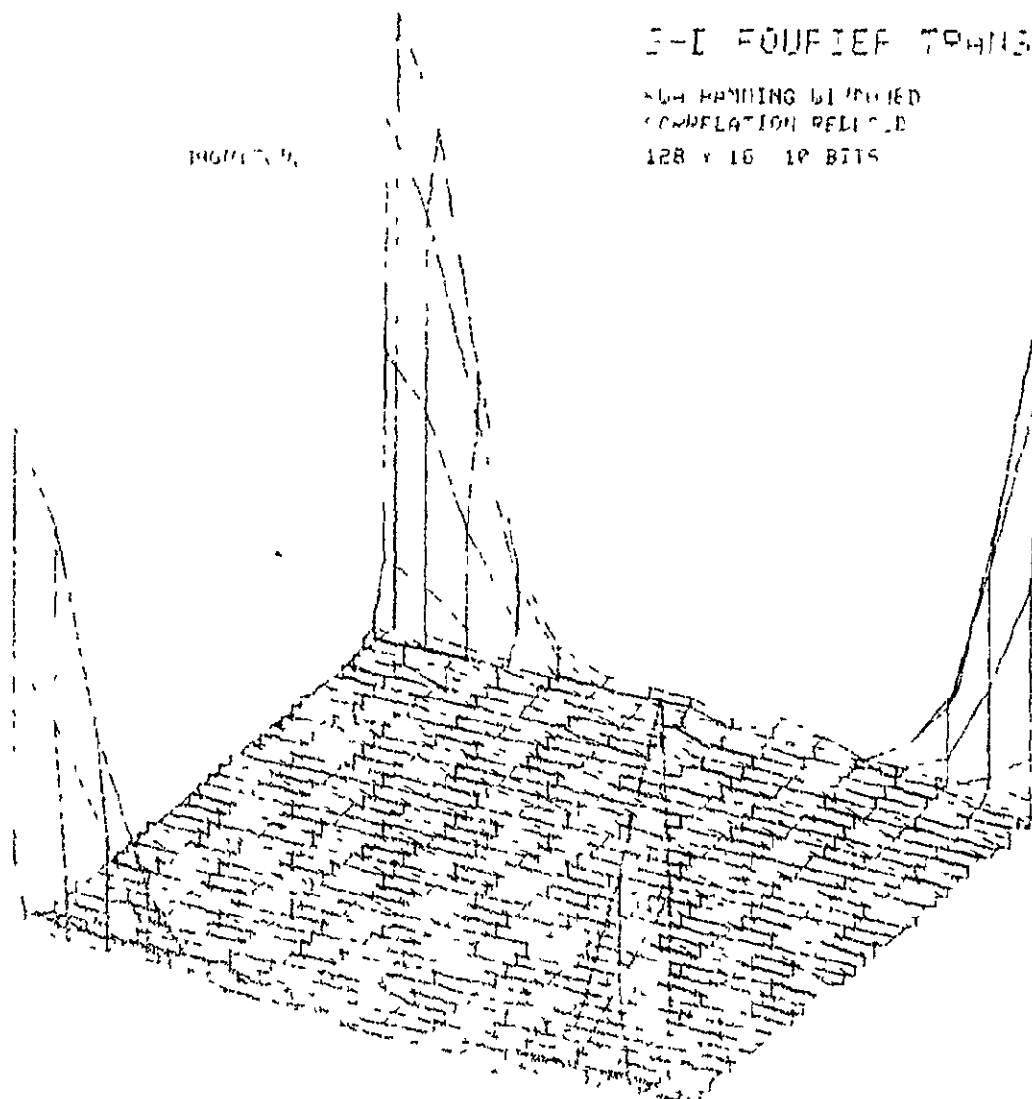
POWER SPECTRUM

2-D FOURIER TRANSFORM

NO HANNING WINDOW

CORRELATION RELIABLE

128 x 16 10 BITS



HORIZONTAL

C. T. Dreier
11-1-76 UTM

Figure 2-4. Facial Power Spectrum (16 x 128).

Facial Scene Power $(g(i,j)^2)^{1/2}$ for 40 Scenes

12

ORIGINAL PAGE IS
OF POOR QUALITY

13

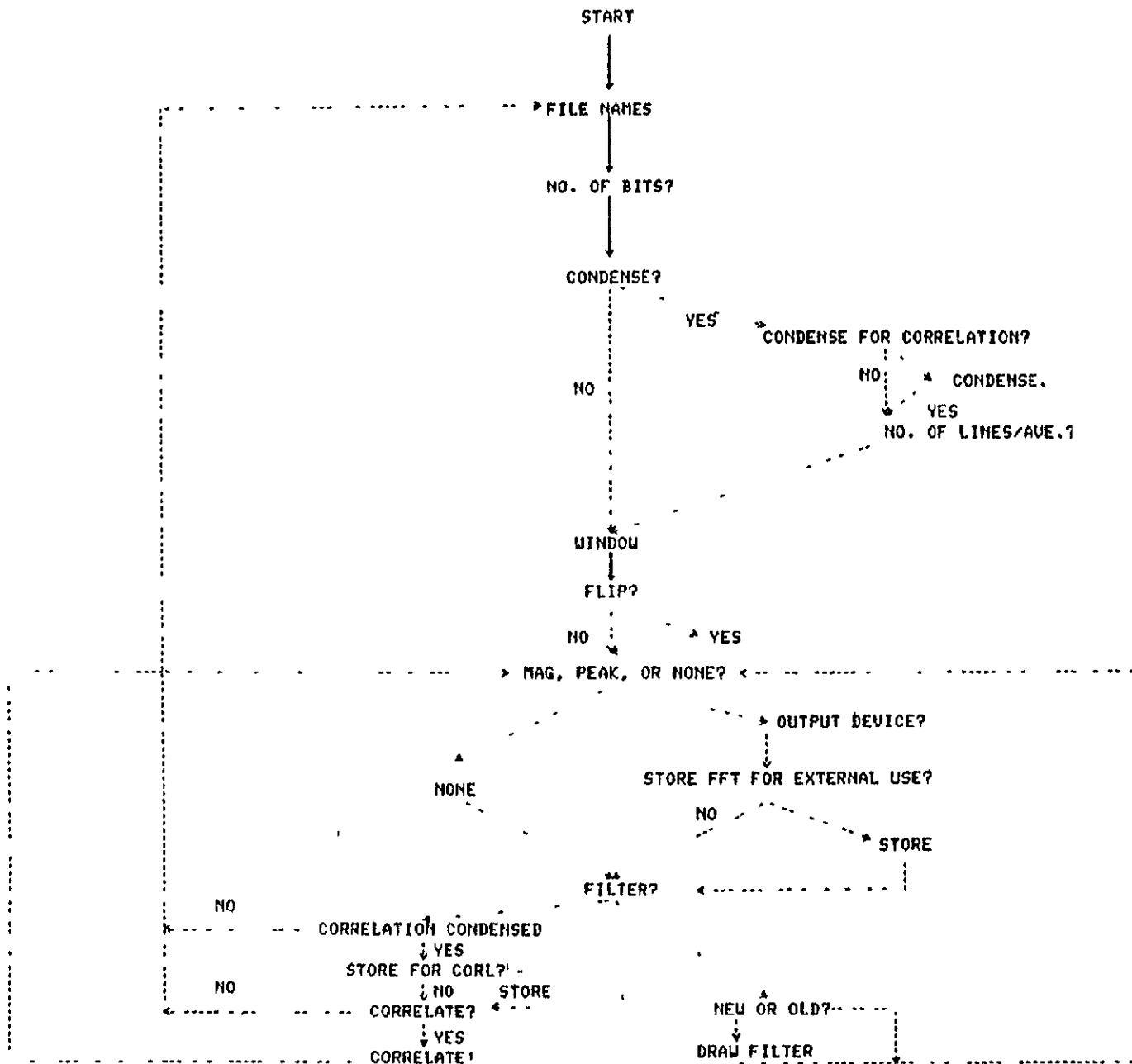


Figure 2-5. System Flow Chart.

		BIT PATTERN N = 10									
DECIMAL VALUE											
1023		1	1	1	1	1	1	1	1	1	1
797		1	1	0	0	0	1	1	1	0	1
312		0	1	0	0	1	1	1	0	0	0
123		0	0	0	1	1	1	1	0	1	1

		N = 4									
15		0	0	0	0	0	0	1	1	1	1
12		0	0	0	0	0	0	1	1	0	0
4		0	0	0	0	0	0	0	1	0	0
1		0	0	0	0	0	0	0	0	0	1

VALUES SHIFTED 6 POSITIONS TO THE RIGHT

Figure 2-6. Bit Reduction

3. CONDENSE AND/OR CONDENSE FOR CORRELATION - As has been previously mentioned, to perform correlation, the data must be zero filled in each direction to prevent wrap-around errors. A scene of 8 x 230 lines is first augmented with zeros to 8 x 256 to establish a modulo 2 radix in each direction. It is then augmented to 16 x 256 to provide zeros in the horizontal direction. Since the array handling size of the computer was effectively limited to 2^{11} complex data points, it was necessary to average every four lines together to reduce the matrix before zero augmenting in the vertical direction. (As mentioned in the discussion on information bandwidth, the effect of averaging is negligible.) The data was then vertically augmented to 16 x 128. See Fig. 2-7a.

The option is also provided to further reduce the data in the vertical direction by averaging together a specified number of lines. For example, specifying eight lines/average after the data has already been zero augmented for correlation would result in the data structure shown in Fig. 2-7b.

4. WINDOW - A choice of three window functions is provided [13]:

- (1) Square window

- (2) Hamming window, $w(t) = .54 + .46 \cos \frac{2\pi t}{T} \quad |t| < \frac{T}{2}$

- (3) Hanning window, $w(t) = .5 + .5 \cos \frac{2\pi t}{T} \quad |t| < \frac{T}{2}$

It should be noted that the windows are placed over only the measured data - not the zero-augmented sections.

5. FLIP - An option is provided to perform the operation $\hat{t}(x,y) = t(-x,-y)$. Note that correlation performed without this operation results in two-dimensional convolution, $F^{-1}(G(f_x, f_y)T(f_x, f_y)) = g(x,y)*t(x,y)$. After the FLIP option, the DFFT is taken on the available data base.
6. MAC, PEAK, OR NONE - An output listing can be obtained of the entire power spectrum (MAGnitude), the highest value of the magnitude (PEAK), or the output can be suppressed (NONE). The scene power is also listed.

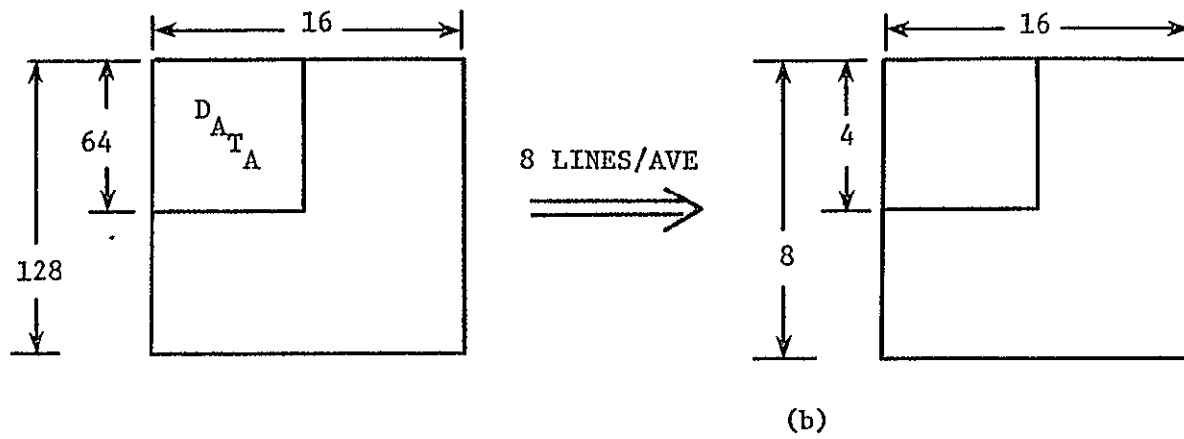
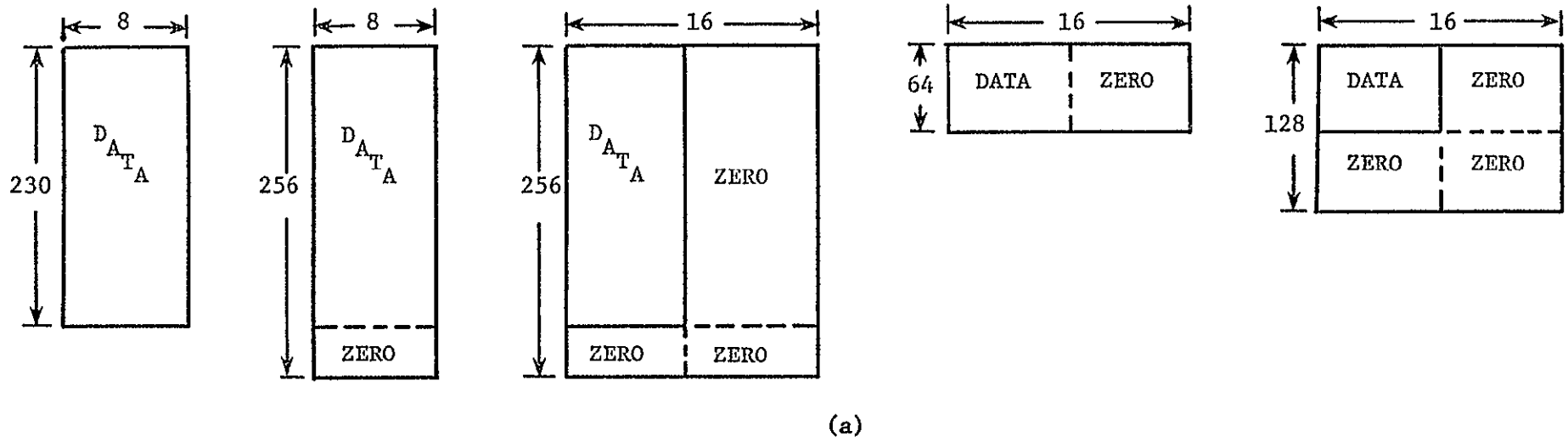


Figure 2-7. Augmenting and Condensing Data for Correlation.

7. STORE FOR EXTERNAL USE - The transformed data can be stored under an arbitrary file name for later use.
8. FILTER - An option is provided to draw (via a Tektronix 4014-1 graphics interface) any two-dimensional filter. The filter's magnitude components are drawn in the frequency domain and then stored for future use. An example of a two-dimensional filter is shown in Fig. 2-8 and 2-9. Note that the filter is symmetric in frequency about the foldover point.
9. STORE FOR CORRELATION - The transformed data is stored in a reserve file to be used in all subsequent correlations and convolutions as a template.
10. CORRELATE - Correlation is performed by multiplying the stored transform (template $T(f_x, f_y)$ or $T(f_x, f_y)$) with the current transform $G(f_x, f_y)$ and then performing an inverse transform.
11. CORRELATION CONDENSED - This is a non-optional branch based upon whether or not the data was zero augmented for correlation. If not augmented, no correlation is performed; and the program returns to the sequence FILE NAME.

Example

1. Establish a template. (In all tests file AA1 was used as the test template. See Fig. 2-10 for a listing of the template-forming sequence.) At this point, a file is stored which consists of a scene (8 x 230) augmented and reduced to 16 x 16, flipped, and transformed.

2. Establish a sequence for template matching (Fig. 2-11). All solid lines indicate an established flow chart path, with the branch decisions printed next to the branch questions.

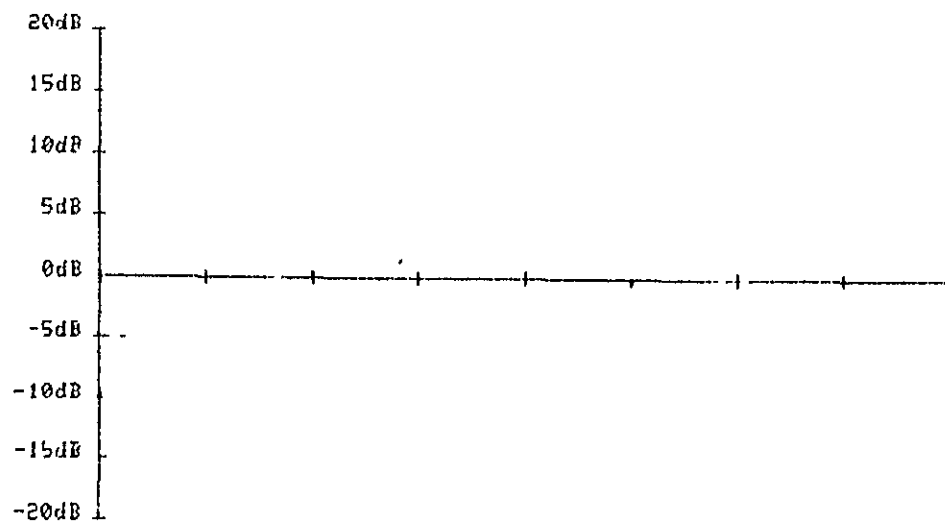
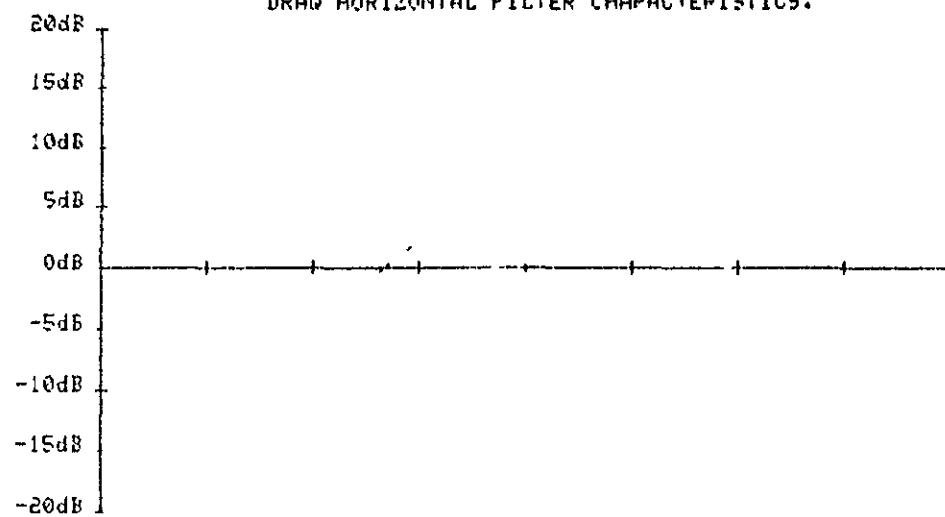
3. Enter a list of files to be used in the template matching procedure.

A typical output appears in Table 2. PWTOT is the scene power, while XHI is the peak value of the correlation.

Results

A brief note is necessary concerning the scene data. Scenes AA1 through AA5 are all facial pictures of the same person, taken at different times. The

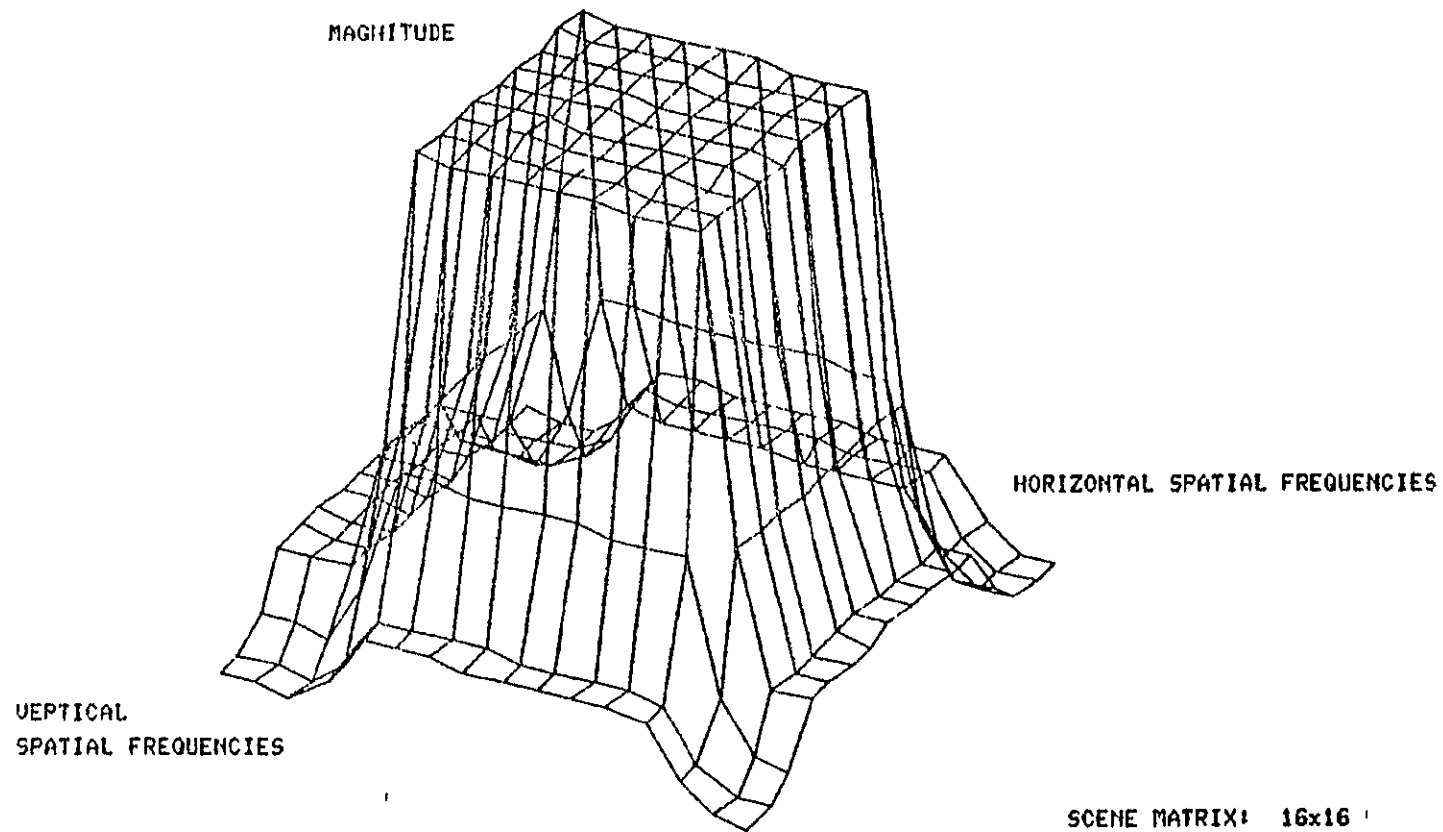
DRAW HORIZONTAL FILTER CHARACTERISTICS.



WRITE VERTICAL FILTER CHARACTERISTICS.

Figure 2-8. Two Dimensional Filter Characteristic Projections.

TWO DIMENSIONAL FILTER CHARACTERISTICS FREQUENCY DOMAIN



C.T.Droher
12-14-76 UUN

Figure 2-9. Two Dimensional Filter, Frequency Domain.

FILE NAME?
HA1
NO. OF BITS? (1=10)
10
CONDENSE?
Y
CONDENSE FOR CORRELATION? (IF YOU DO NOT DESIRE
TO CONDENSE, BUT WISH TO CORRELATE, TYPE 'Y' FOR YES, AND THEN
ENTER A 1 FOR NUMBER OF LINES PER AVERAGE.)
Y
OF LINES PER AVERAGE?
3
WINDOW: 1=SQUARE, 2=HAMMING, 3=HAMMING.
1
FLIP?
Y
MAGNITUDE OUTPUT, PEAK POINT, OR NONE, (M,P,N)
N
FILTER?
N
STORE FFT FOR CORRELATION?
Y
CORRELATE?
N
FILE NAME?
DONE
.
@

Figure 2-10. Template Forming Sequence.

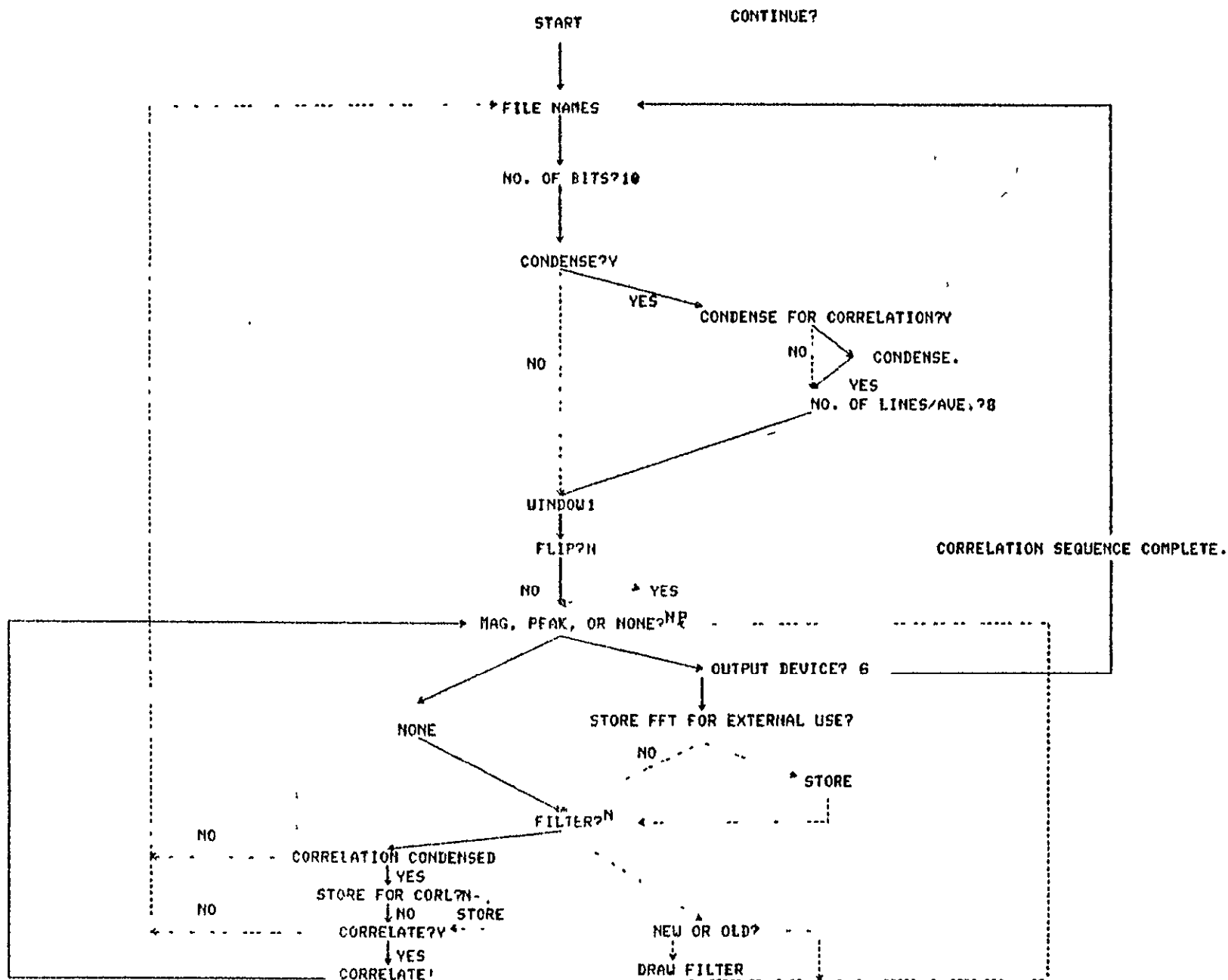


Figure 2-11. Flow Chart Training Sequence.

Intermediate Output

JJS1 (AA1)

1 BITS	CORR. COND?	LINES/AVE	WINDOW	FILTER?	PWTOT	XHI
10	Y	8	1	N	5089.6904	5089.6914
10	Y	8	1	N	5083.7793	5089.5762
10	Y	8	1	N	5106.3447	5089.5215
10	Y	8	1	N	5116.9551	5089.4414
10	Y	8	1	N	5117.9023	5089.3604
10	Y	8	1	N	5514.4854	5073.4668
10	Y	8	1	N	5540.0029	5073.0771
10	Y	8	1	N	5544.0469	5073.4121
10	Y	8	1	N	5545.9824	5073.0889
10	Y	8	1	N	5517.5703	5074.0771
10	Y	8	1	N	5498.1279	5074.3711
10	Y	8	1	N	5496.8848	5075.0762
10	Y	8	1	N	5520.9482	5074.1221
10	Y	8	1	N	5535.4053	5073.4385
10	Y	8	1	N	5533.9707	5073.7480
10	Y	8	1	N	5276.5439	5066.8555
10	Y	8	1	N	5373.0781	5079.1094
10	Y	8	1	N	5378.2656	5079.4326
10	Y	8	1	N	5379.6807	5075.3193
10	Y	8	1	N	5403.1885	5076.4043
10	Y	8	1	N	5402.7236	5078.6270
10	Y	8	1	N	5444.0879	5079.0234
10	Y	8	1	N	5443.8291	5078.9687
10	Y	8	1	N	5444.0176	5078.3379
10	Y	8	1	N	5458.0986	5078.7969
10	Y	8	1	N	5778.9033	5047.7354
10	Y	8	1	N	5784.4375	5046.9355
10	Y	8	1	N	5799.3320	5044.3105
10	Y	8	1	N	5796.5420	5043.0586
10	Y	8	1	N	5618.2529	5081.7568
10	Y	8	1	N	5576.2373	5079.3525
10	Y	8	1	N	5601.2109	5079.8613
10	Y	8	1	N	5632.8457	5081.7168
10	Y	8	1	N	5635.1475	5081.5215
10	Y	8	1	N	5500.3447	5012.7930
10	Y	8	1	N	5469.0273	5019.4707
10	Y	8	1	N	5461.3760	5021.7598
10	Y	8	1	N	5459.4414	5018.3887
10	Y	8	1	N	5425.5361	5019.2129

subject was allowed to change his expression and move about but was instructed to stay within the camera window and focus and to display only frontal views.

Refer to Table 2. The correlation peaks (XHI) associated with scenes AA1-AA5 are very tightly grouped about a value of 5089. All remaining scenes are below this value. A threshold of 5089 would generate a simple discriminant function capable of separating a correct identification (i.e., a match of person AA1 with any of his other facial scenes) from the remaining scenes with complete accuracy over the test data. See Fig. 2-12. Similar results are obtained using (1) reduced word size and (2) reduced data matrix. In all cases it was possible to obtain a linear discriminant function that would give 100 percent separation over the test data.

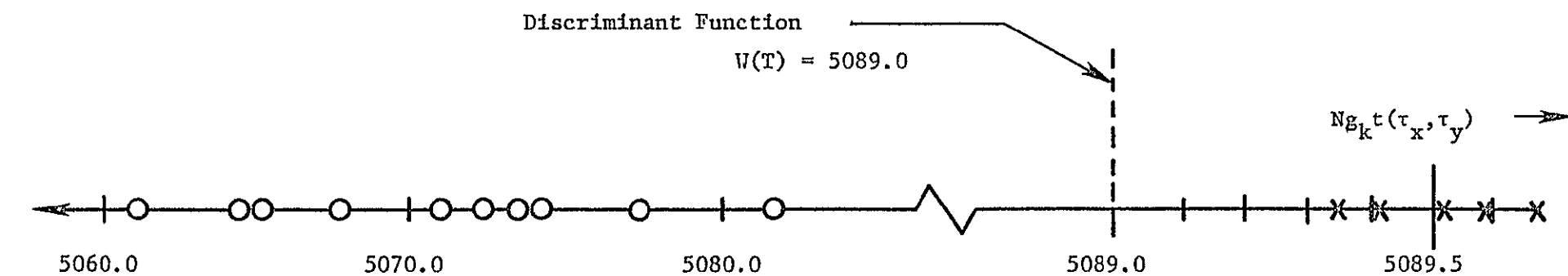
In order to evaluate the system performance a performance index (PI) was defined as

$$\frac{\min \text{ over } w_1 N_{g_k t}(T_x, T_y) - \max \text{ over } w_2 N_{g_k t}(T_x, T_y)}{\min \text{ over } w_1 N_{g_k t}(T_x, T_y)} \times 1000$$

This index measures a normalized distance between the clustered correct correlation peaks and the closest mismatched peak.

The effect of varying the bandwidth (lines/average) and word size is shown in Table 3. Three significant aspects are immediately apparent:

1. For large word sizes, the PI peaks at lines/average = 2 and then decreases markedly. Specifying two lines/average reduced the scene to 16 x 64 data pairs and allows a vertical spatial frequency resolution of 32 harmonics. It was pointed out previously that the power spectrum leveled off after the 32nd harmonic and thus agrees well with the concept of information bandwidth.
2. The PI is fairly constant over the values of 1 to 4 for lines/average. While a peak is obtained at 2, using half again as much data only slightly reduces the PI. Accurate results could thus be obtained with reduced system complexity and increased throughput.



Template: AA1
 X = Scenes AA1-AA5
 O = Other Scenes

Figure 2-12. A Linear Discriminant

TABLE 3

Word Size versus Bandwidth

Bits	Lines/Average						
	1	2	4	8	16	32	64
10	1.927	1.940	1.850	1.502	1.196	.6726	.3623
9	1.939	1.944	1.853	1.505	1.1976	.6738	.3634
8	1.950	1.953	1.860	1.517	1.201	.8172	.279
7	1.964	1.9692	1.876	1.526	1.217	.6871	.184
6	2.001	2.004	1.918	1.573	1.250	.7129	.3833
5	2.048	2.070	1.983	1.639	1.304	.7522	.313
4	2.278	2.308	2.212	1.905	1.427	.8402	.365
3	4.300	4.283	4.003	3.427	2.719	1.731	1.559
2	13.88	13.816	13.43	10.47	7.88	4.703	1.939
1	47.968	47.502	47.152	38.14	30.167	15.019	5.23

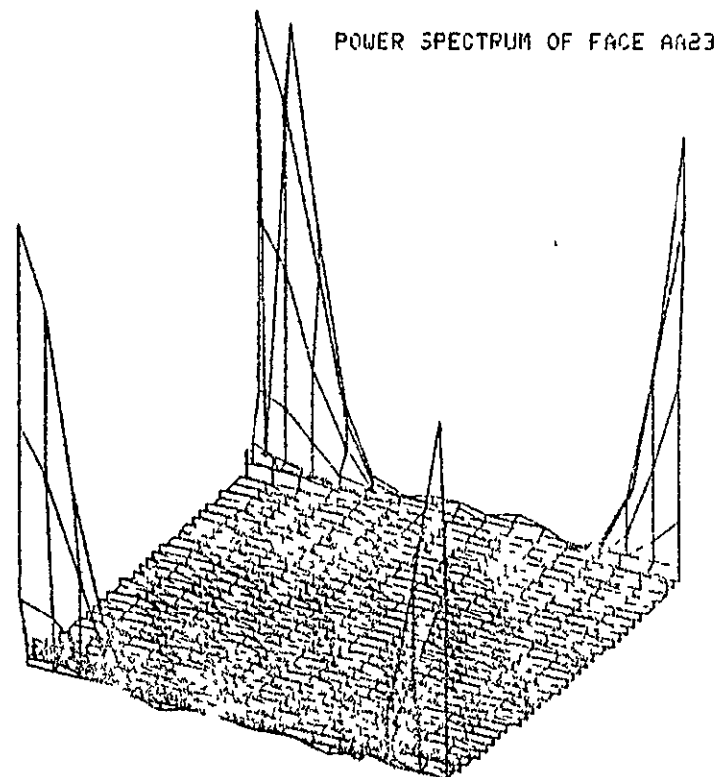
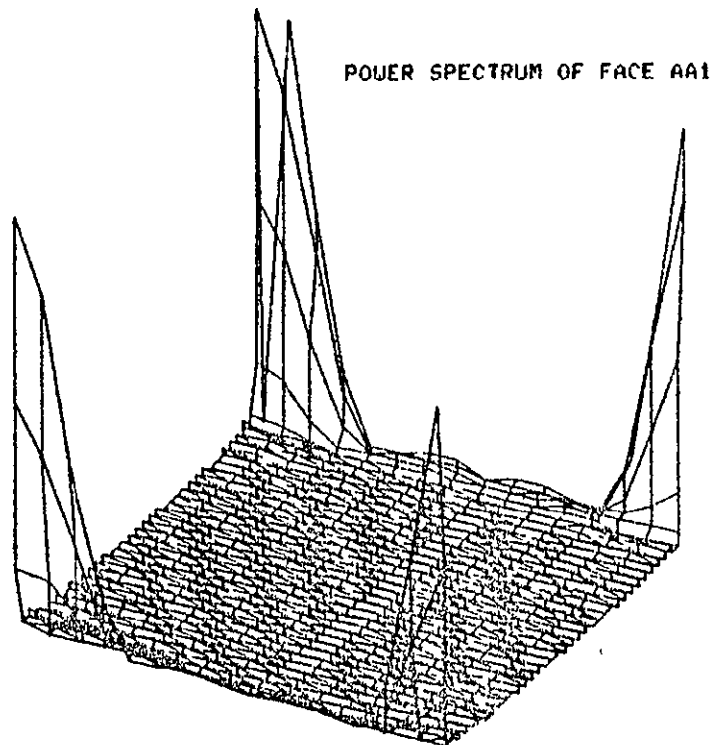
3. A large increase in the index results as the word size is reduced. This effect has immediate cost benefits:
 - a. Since excellent recognition can be achieved with only one bit, an inexpensive high-speed comparator can be substituted for a slower, more costly, 10-bit A/D. Also, the noise benefits of large quantization levels are realized and thus loosen many potentially expensive noise reduction criteria.
 - b. In terms of storage a typical 16-bit word could hold two lines of video at eight samples/line, 1-bit samples. An entire condensed array would fit into 256 words (128 x 16 data pairs (complex data) = 4048 bits = 256 16-bit words.) The cost reduction is obvious.
 - c. The Boolean algebra associated with single-bit multiplication and summation can be easily and quickly handled with hardware devices. It is conceivable that the entire process could be carried out in near real-time circumstances, with a 16-bit mini- or high-speed 8-bit microcomputer containing only limited core.

The above three points are worth repeating. They point the way to an inexpensive, fast, and highly reliable pattern recognition system that can be fabricated with existing technology. Using charge-coupled devices would even further increase the system speed, although at an unknown cost index (measured against existing TTL devices) and availability.

Filtering

Another aspect worth mentioning is that of high frequency variance. Consider the power spectrums pictured in Fig. 2-13. They appear remarkably similar, especially at the high-power, low-frequency terms. This is intuitively appealing, since faces, after all, are quite similar. (General shape, two eyes, a nose, mouth, etc., are common features.) Apparently it is small, high spacial frequencies which contribute significantly to the differences.

To test this theory a high-pass filter was implemented (Fig. 2-8 and 2-9.) The result of this filter on various word sizes for eight lines/average



POWER SPECTRUM COMPARRISON
OF TWO DIFFERENT FACES.

(SCENES CORRELATION REDUCED TO 16x128,
HAMMING WINDOWED.)

NOTE: DC TERM SUPPRESSED

C.T. Drehan
12-17-76 UVA

Figure 2-13. Power Spectra

is tabulated in Table 4. Apparently, facial scene differences are contained in the upper spatial frequencies. Analog preprocessing of the incoming video signal could thus significantly improve the system's performance index.

It should be pointed out, however, that, if high frequencies are to be retained, the sampling frequency must remain high and the data matrix must remain large enough to resolve the higher order components. The price of increasing the PI via this technique is thus paid for with higher speed devices and larger memory requirements.

CCD Implementation

A proposed facial recognition system employing charge-coupled devices to perform transform operations is outlined in Fig. 2-14 [15]. Note that the data presented to the camera now becomes the template and not the stored scene.

If a pattern $G_k(f_x, f_y)$ is presented and results in a "0" or mismatched output, the scene pointer is indexed to the $K+1$ scene and the process repeated. Any successful match (a "1" output) stops the process, and access (in a security system, for example) is allowed. Access is denied if no match is achieved.

Summary

Recognition of facial scenes can be achieved with high accuracy using normalized cross correlation via transforms. Detailed investigation of spatial frequencies reveals that accurate identification can be achieved using low sampling rates at reduced bandwidths and small array sizes. Performance can be improved considerably using coarsely quantized gray levels and preprocessed data, although the latter imposes some restrictions.

Such "brute-force" technique for pattern recognition may now find practical applications because of new CCD devices like the Chirp z -transform circuit. Although the system described above could be implemented using all digital technology, the overall advantages of a CCD system make it far more attractive.

TABLE 4

Various Word Sizes for Eight Lines/Average

<u>Word Size (Bits)</u>	<u>Performance Index</u>
10	58.29
9	58.53
8	58.67
7	58.05
6	58.57
5	59.30
4	72.10
3	96.91
2	243.00
1	513.3

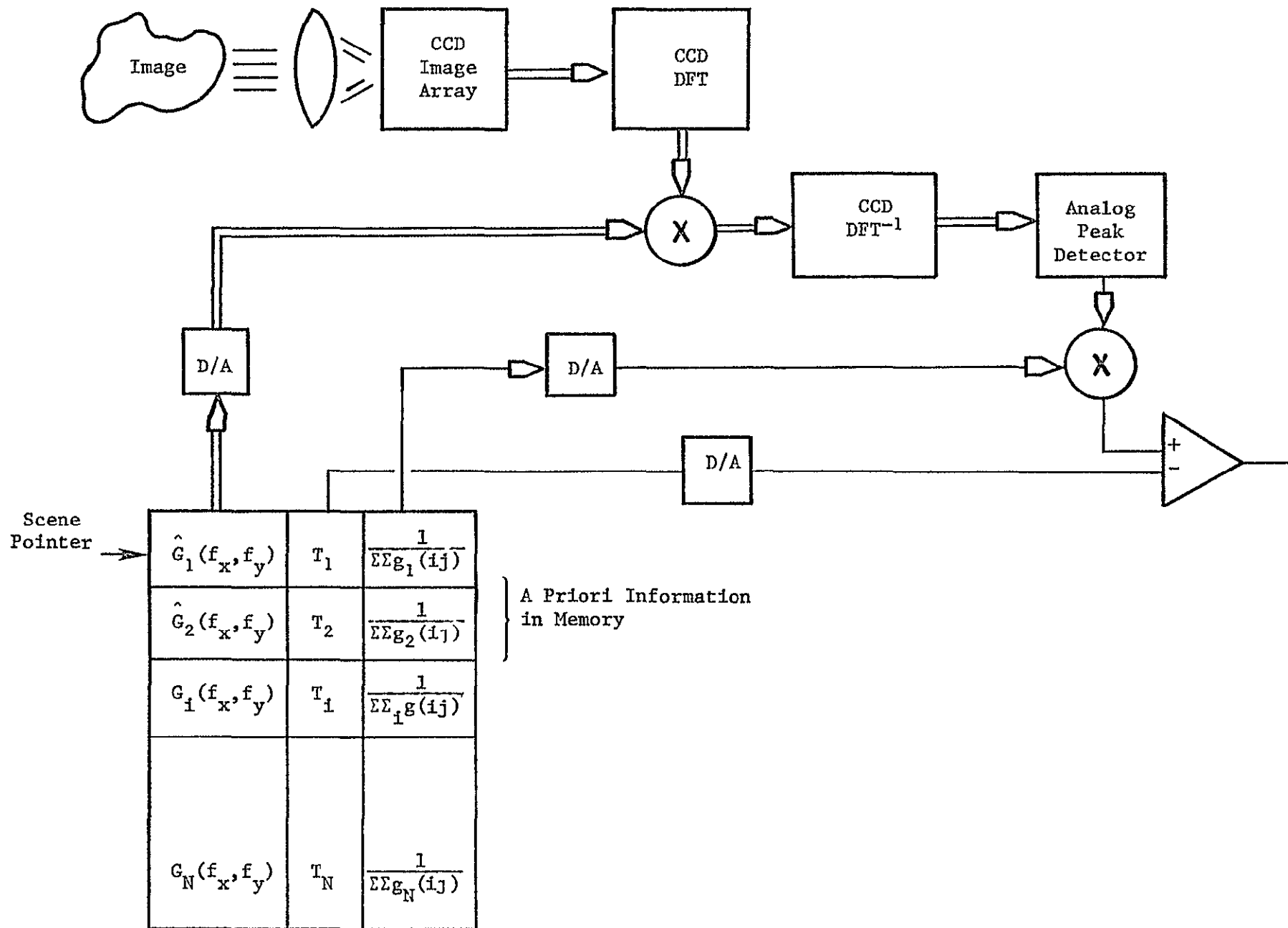


Figure 2-14. Proposed CCD Implementation

BIBLIOGRAPHY

1. N. Ahmed, T. Natarajan, K. R. Rao, "Discrete Cosine Transform," IEEE Trans. on Comp., pp. 90-93.
2. Charles M. Rader, "An Improved Algorithm for High Speed Autocorrelation and Application to Spectral Estimation," Digital Sig. Processing, p. 330, IEEE Press, N.Y., 1972.
3. Glenn D. Bergland, "Fast Fourier Transform Hardware Implementations - An Overview," Digital Sig. Processing, pp. 349-53, IEEE Press, N.Y., 1972.
4. Charles M. Puckette, "Bucket Brigade Transversal Filters," IEEE Trans. on Comm., Vol. COM-22, No. 7, pp. 926-34, July 1974.
5. L. R. Rabiner, "The Chirp z-Transform Algorithm," Digital Sig. Processing, p. 322, IEEE Press, N.Y.
6. Dennis R. Buss, Dean R. Collins, Walter H. Bailey, C. Richard Reeves, "Transversal Filtering Using Charge Transfer Devices," IEEE Jnl. of Sld. St. Cir., Vol. SC-8, No. 2, pp. 138-46, Apr. 1973.
7. Robert W. Brodersen, C. Robert Hewes, Dennis D. Buss, "A 550-State CCD Transversal Filter for Spectral Analysis," IEEE Jnl. of Sld. St. Cir., Vol. SC-11, No. 1, pp. 75-84.
8. Miles A. Copeland, Dipak Roy, John E. Beynon, Frank Dea, "An Optical CCD Convolver," IEEE Jnl. of Sld. St. Cir., Vol. SC-11, No. 1, pp. 84-87.
9. G-AE Subcommittee on Measurement Concepts, "What is the Fast Fourier Transform?" Digital Sig. Processing, p. 243, IEEE Press, N.Y.
10. G. D. Bergland, "A Guided Tour of the Fast Fourier Transform," Digital Sig. Processing, pp. 232-37, IEEE Press, N.Y.
11. Peter E. Blankenship, E. M. Hofstetter, "Digital Pulse Compression via Fast Convolution," IEEE Trans. on Acous., Speech, & Sig. Processing, Vol. ASSP-23, No. 2, pp. 189-201, Apr. 1975.
12. Richard O. Duda, Peter E. Hart, Patt. Class. & Scene Anal., pp. 279-80, John Wiley & Sons, N.Y., 1973.
13. William D. Stanly, Digital Sig. Processing, pp. 239-56 & 226-27, Reston Publ. Co., Inc., Reston, Va., 1975.
14. Lloyd W. Martinson, Richard J. Smith, "Digital Matched Filtering and Pipelined Floating Point Fast Fourier Transforms (FFT's)," IEEE Trans. on Acous., Speech, & Sig. Processing, Vol. ASSP-23, No. 2, pp. 222-26, Apr. 1975.

BIBLIOGRAPHY (continued)

15. Gault A. Antclifte, L. J. Hornbeck, W. W. Chan, J. W. Walker, Walden C. Rhines, Dean R. Collins, "A Backside Illuminated 400 x 400 Charge-Coupled Imager," IEEE Trans. on Electron Devices, Vol. ED-23, No. 11, pp. 9225-1232, Nov. 1976.
16. E. A. Parrish, Jr., E. S. McVey, "Implications of Charge-Coupled Devices for Pattern Recognition," IEEE Trans. on Comp., Vol. C-25, No. 11, pp. 1146-52, Nov. 1976.
17. T. G. Stockham, Jr., "High-Speed Convolution and Correlation," Digital Sig. Processing. p. 330. IEEE Press, N.Y., 1972.

III. A PROGRAMMABLE DISCRETE ANALOG PROCESSOR FOR PATTERN RECOGNITION

This project in the area of discrete analog processing involves the design and implementation of a high-speed unit capable of performing numerous transversal filtering operations on a fixed record of analog input samples. Such a system is necessary because CCD filters are not yet commercially available. The design has been executed using available off-the-shelf logic and discrete analog components to attain speed and flexibility of usage. At the same time, some sacrifices in package count and circuit integrability have been made. The unit under construction is thus not a prototype but a tool for studying the properties and requirements of a class of similar systems which will employ CCDs. In addition, it also allows investigation of interface requirements for microcomputer control of CCD-type processors.

The main subsystems are a programmable sequential filter (PSF) and a signal comparator. Of these, the former is far more complex and has received exclusive attention thus far. Both subsystems run as slaves to a microprocessor-based controller which, during operation, performs overhead functions periodically and, during test and development, provides a flexible man/machine interface.

The functional organization of the PSF subsystem is shown in Fig. 3-1. The microcomputer controls the PSF via an 8-bit status/command latch. In return it monitors the filter's computational completion status via an interrupt line. Digital logic regulates the loading (sampling) of an input analog signal with respect to number of samples and possible input source multiplexing, the necessary data being encoded in ROM. The samples are shifted into a bucket-brigade analog shift register with broadside readout capability. Other logic controls the subsequent computing phase in which the various transversal filter outputs are developed sequentially as analog outputs. That is, for an input vector x and a transformation matrix A the components of Ax are generated. The structure of the computation (i.e., dimensions and possibly the sparseness of A) is once more ROM-encoded, while the specific matrix elements are storable with 8-bit accuracy in a coefficient RAM under microcomputer control.

To process the input sample vector x its i components are read out of the broadside shift register taps through a fast analog multiplexer and presented sequentially as reference input to a multiplying D/A converter (MDAC). Simultaneously, the filter coefficients are read from RAM, and analog product outputs are generated which are summed by a resettable integrator. Data throughput is enhanced by overlapping the multiplication and summation operations. The readout process is repeated until all of the required filter outputs have been generated.

One feature thought to be desirable in a non-dedicated processor is automatic ranging of the input signal to the internal signal range most compatible with linearity of the analog memory device and maximum overall S/N ratio. The PSF analog input circuitry is capable of shifting and amplifying the input to achieve the desired internal signal range. The offset and gain required are stored as signals on high-impedance gates and are updated periodically by the response of feedback control loops to inputs representing maximum and minimum input signal levels. In addition, the PSF is designed to periodically store a declared numerical zero signal level on a holding gate in order to provide flexible four-quadrant capability. The tasks of controlling the autoranging and numerical-zero-acquisition circuitry are the "overhead" tasks previously mentioned which fall to the microcomputer. To date, the following steps in development of the PSF have been accomplished:

1. Completed design, construction, and test of microcomputer and random logic control circuitry.
2. Test and measurement of input/output characteristics of the selected analog memory device.
3. Design, breadboard, and test of auto-ranging control loops and sample/hold circuits.
4. Breadboard of multiplier and resettable integrator circuitry.

Final assembly on circuit cards of the entire PSF is presently nearing completion. Remaining work includes in-circuit testing of the analog multiplexer, multiplier, and integrator and verification of control timing relationships and computational functions.

When completed, it is intended that this system will serve as a general-purpose OP-SAP emulator. It will be used as a feature extractor and as a classifier in several different pattern recognition system studies.

IV. DISCRETE WORD RECOGNITION

Introduction

The potential of CCD technology for performing fast and complex processing of speech signals has generated interest in the possibility of voice-actuated command and control systems for aviation. The system envisioned would be capable of recognizing discrete words spoken by an operator for whom it had been previously trained. The commands would include such things as requests about altitude, range, speed, etc. Described below is an investigation involving a theoretical study of the application of certain CCD devices to signal processing techniques used in automatic speech recognition. Discrete word recognition systems generally use the same approach as any other pattern recognition system. This process includes the measurement of the input pattern, the creation of features from these measurements, and the identification of the pattern using the features. A slight shift in emphasis, however, may occur in the implementation of the two systems. In general, the major area of interest in discrete word recognition systems is the feature extraction process while using a relatively simple classification scheme. This may or may not be the case for a typical pattern recognition system. Since the classification scheme in a word recognition system is simple, a good feature set is imperative.

Schafer and Rabiner [1] report that both time domain and frequency domain feature sets can be used for speech recognition. Popular time domain methods, however, still result in a spectral estimate of the speech waveform. Ichikawa [2] did a study on various parameter sets for speech recognition. Fig. 4-1 shows the relationship of these parameter sets. Ichikawa determined that of the four different parameter sets which he tested, the cepstral envelope technique yielded the best results.

A familiar method for feature extraction from speech data is to employ multiple bandpass filters to generate a rough spectral envelope. Hatan [3] describes a speech recognition system for a numerical control language. Feature extraction was performed by 24 bandpass filters over a range of 110-7000 Hz. A recognition rate of 98% was achieved on 26 word

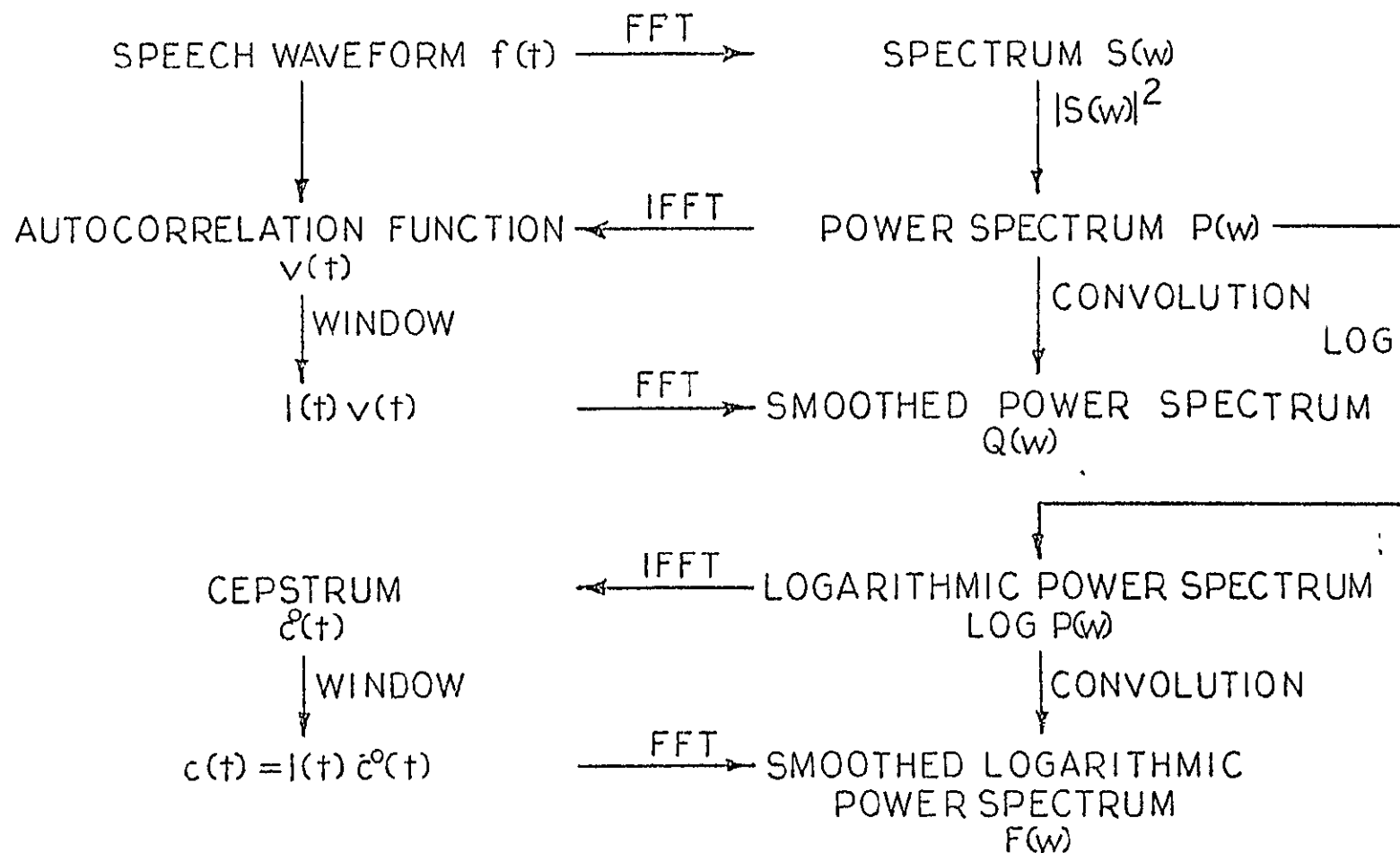


Figure 4-1. Relationship of Various Parameter Sets
for Speech Recognition [6:22].

vocabulary having a well-defined syntactical structure. This system also boasted real time operation capability.

It was concluded from reports such as described above that the spectral envelope of a speech waveform was generally required in the feature extraction process. The spectral envelope may be used either exclusively as a parameter set, as a part of a parameter set, or to derive a parameter set. It was, therefore, decided to investigate the application of the Fourier transform CCD processor to speech waveforms. The actual device of interest is the power spectrum processor which uses the sliding Chirp z-transform algorithm. The possible availability of such a device in the near future was a large factor in the choice of the methodology [4].

The investigation was divided into two phases. The first phase involved the computer simulation of the CCD power spectrum processor. The current lack of availability of the device made the simulation necessary. Since the CCD device implements the Chirp z-transform with a sliding input data window, it was also necessary to generate the power spectrum via a fast Fourier transform routine to verify the results. The second phase included processing actual speech signals by the simulation routines. A routine used to determine the spectral content of an utterance from short time segments was also simulated using CCD technology and tried on actual speech data.

CCD Power Spectrum Device Simulation

The charge-coupled device implementing either a discrete Fourier transform or a power spectrum uses the Chirp z-transform algorithm [5]. The discrete Fourier transform is defined as

$$x_k = \sum_{n=0}^{N-1} x_n e^{-2\pi i n k / N} \quad (4-1)$$

By making the substitution

$$2nk = k^2 + n^2 - (k - n)^2 \quad (4-2)$$

the discrete Fourier transform becomes

$$x_k = e^{-i\pi k^2/N} \sum_{n=0}^{N-1} \left(x_n e^{-i\pi n^2/N} \right) e^{i\pi (k-n)^2/N} \quad (4-3)$$

This equation is known as the Chirp z-transform (CZT) equation for the discrete Fourier transform. The CZT consists of three basic operations in the following sequence:

- 1) the multiplication of a complex input x_n by a complex chirp waveform,
- 2) the convolution of this premultiplication and a complex chirp waveform, and
- 3) the multiplication of the convolution result and another complex chirp waveform.

When the power spectrum is desired the postmultiplication is not required. The equation for the power spectrum is

$$|x_k|^2 = \left| \sum_{n=0}^{N-1} (x_n e^{-i\pi n^2/N}) e^{i\pi (k-n)^2/N} \right|^2 \quad (4-4)$$

For the case of real input signals, (4) can be expanded to become

$$\begin{aligned} |x_k|^2 = & \left[\sum_{n=0}^{N-1} (x_n \cos \pi n^2/N) \cos \pi (k-n)^2/N \right. \\ & \left. + \sum_{n=0}^{N-1} (x_n \sin \pi n^2/N) \sin \pi (k-n)^2/N \right]^2 \\ & + \left[\sum_{n=0}^{N-1} (x_n \cos \pi n^2/N) \sin \pi (k-n)^2/N \right. \\ & \left. - \sum_{n=0}^{N-1} (x_n \sin \pi n^2/N) \cos \pi (k-n)^2/N \right]^2 \end{aligned} \quad (4-5)$$

The power spectrum equation is implemented using multiplying digital-to-analog converters for the premultiplication and transversal filters for the convolution having impulse responses of

$$h_m^{\sin} = \sin \pi m^2/N \quad , m = 0, 1, 2, \dots, N - 1 \quad (4-6)$$

and

$$h_m^{\cos} = \cos \pi m^2/N \quad , m = 0, 1, 2, \dots, N - 1 \quad (4-7)$$

An N-point power spectrum device would require four transversal filters, each 2N stages long. Also a true CCD power spectrum processor would require hardware to accomplish 50% blanking of the input. These restrictions led to the implementation of the sliding CZT for spectral analysis in CCD devices. The sliding CZT is defined as

$$x_k^s = \sum_{n=0}^{N-1} x_{n+k} e^{-2\pi i n k/N} \quad (4-8)$$

The sliding CZT equation expands as does the true CZT. Each Fourier coefficient, however, is now computed over a slightly different set of input data points. The advantage of using the CZT is that the filters can be N stages long for an N point spectrum and no blanking is necessary. The problem lies in the fact that if the input data is not periodic or stationary, the sliding CZT will not yield the exact spectral coefficients as does the true CZT implementation. A computer simulation package was needed to determine if this restriction would greatly hamper the use of the sliding CZT on speech waveforms.

The simulation package consists of two major routines. The first routine calculates the power spectrum of N data points by a fast Fourier transform algorithm. The other routine calculates the power spectrum using the sliding CZT algorithm. Both routines access input data from digital magnetic tape. The data is stored in single data item records in order to be able to change the spectrum resolution easily. Both routines also contain graphic capabilities for ease of analysis.

The sliding CZT simulation program implements Eq. 4-5. The convolution operation is performed in a subroutine with the premultiplication and sliding data buffer operations being done in the main routine. Data windows

can also be applied to the input data; however, results are only included for rectangular data windows. The CCD simulation routine only includes the mathematics of the device and none of the electrical properties such as transfer inefficiency.

The software packages also includes a data acquisition routine. The routine samples the input at prescribed intervals and for a specified duration and stores them on magnetic tape in a format compatible with the simulation routines. This extra storage medium is necessary for ease of analysis and because the simulation routines are not capable of real time operation. Listings of the simulation routines are found in the Appendix.

Speech Waveform Analysis Using Simulation Package

The real time speech data was acquired in the Computer Systems Laboratory by a Hewlett Packard 2100A computer system. The simulation package was also resident on this computer system. The computer system contains a general purpose data acquisition system capable of acquiring data samples at any rate up to 200 KHz. A block diagram of the system is shown in Fig. 4-2. The sampling process can be initiated by either the computer alone or a combination of the computer and an external synchronization signal. The system also contains selectable low-pass filters to remove aliasing problems. Special circuitry was necessary to acquire speech data using the data acquisition system. A block diagram of the speech interface is shown in Fig. 4-3 [6]. A portion of the interface was devoted to the generation of the sync signal. The speech waveform was passed through an automatic gain control (AGC) circuit to force the signal to occupy the full dynamic range of the A/D converter. Rectification circuitry was also available although it was not used for this investigation.

It is fairly widely accepted that systems for speech recognition can be band-limited to around 4 KHz, thereby requiring at least an 8 KHz. sampling rate. It was also determined that a one-half second interval is sufficient for most utterances of initial interest. These requirements then dictate the processing of approximately 4096 samples per utterance. Because of the large number of samples per utterance, the power spectrum of an entire record was impractical to obtain. A recognized way of

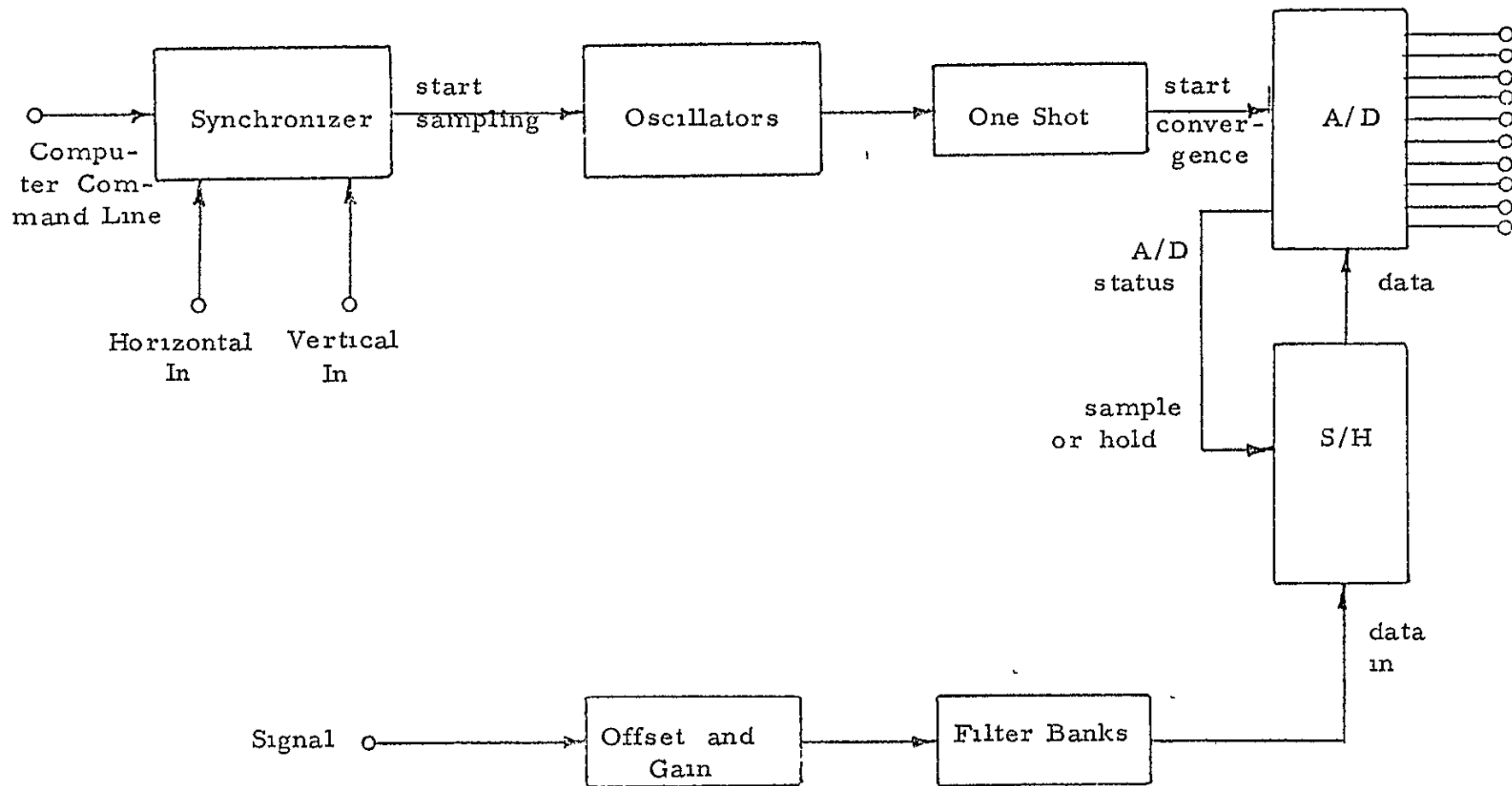


Figure 4-2. Data Acquisition System
Block Program [6:43]

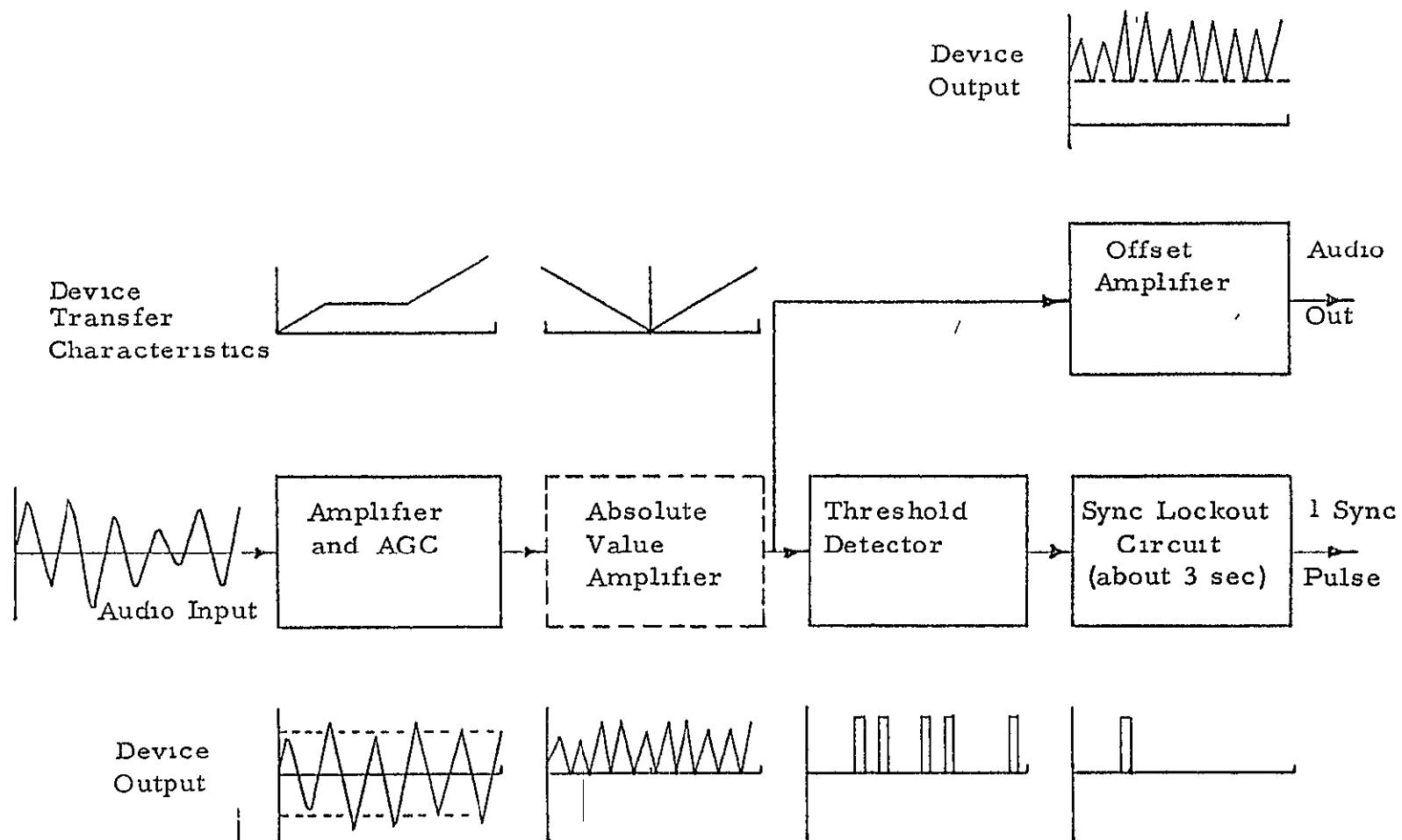


Figure 4-3. Block Diagram of the Speech Interface Box [6:50]

accommodating this amount of data is to break the sampling interval into sub-intervals. The size of each sub-interval is a function of the frequency resolution necessary for correct classification. The power spectrum of each sub-interval is then obtained. This power spectrum is known as a periodogram. A smaller number of periodograms per utterance can be generated by the averaging of adjacent periodograms. These resulting periodograms are then used in most speech recognition systems to create the feature set.

Several one-half second utterances were input to the simulation package. The 4096 data samples were divided into sub-intervals of 128 samples each. This sample size produced a frequency resolution of approximately 62 Hz. Ten adjacent periodograms were then averaged to form a new periodogram in order to reduce the total number of periodograms in the entire utterance. Figs. 4-4 and 4-5 show the results of the first averaged periodogram of the utterance, "Alpha". Fig. 4-4 shows the true power spectrum including the negative frequency terms. Fig. 4-5 shows the result of the CCD simulation with its negative frequency coefficients. The negative frequency coefficients were included for comparison since the sliding CZT algorithm does not guarantee that these coefficients are merely a reflection of the positive coefficients. Figs. 4-6 and 4-7 are similar results of a different utterance.

Conclusions

The results of the simulation show that CCD power spectrum device may find a place in the field of speech recognition. The processing of periodogram averaging tends to cure a major pitfall of the sliding Chirp z-transform algorithm. The sliding CZT processor produces a power spectrum that is very similar to the true power spectrum for segments of speech. CCD devices may possibly rekindle interest in various techniques in speech recognition that were previously put aside for various reasons.

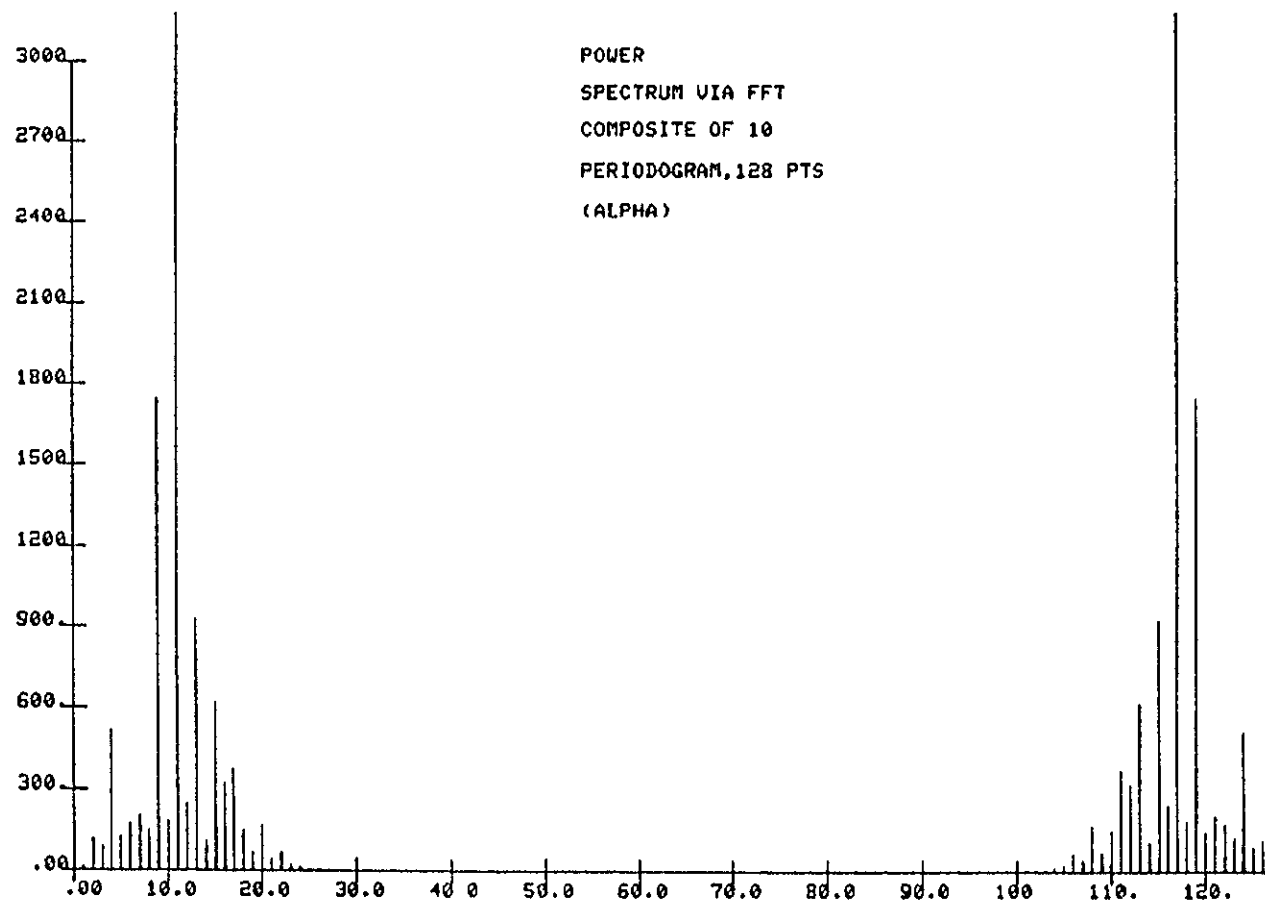


Figure 4-4. True Power Spectrum for the First
Averaged Periodogram of "ALPHA"

ORIGINAL PAGE IS
OF POOR QUALITY

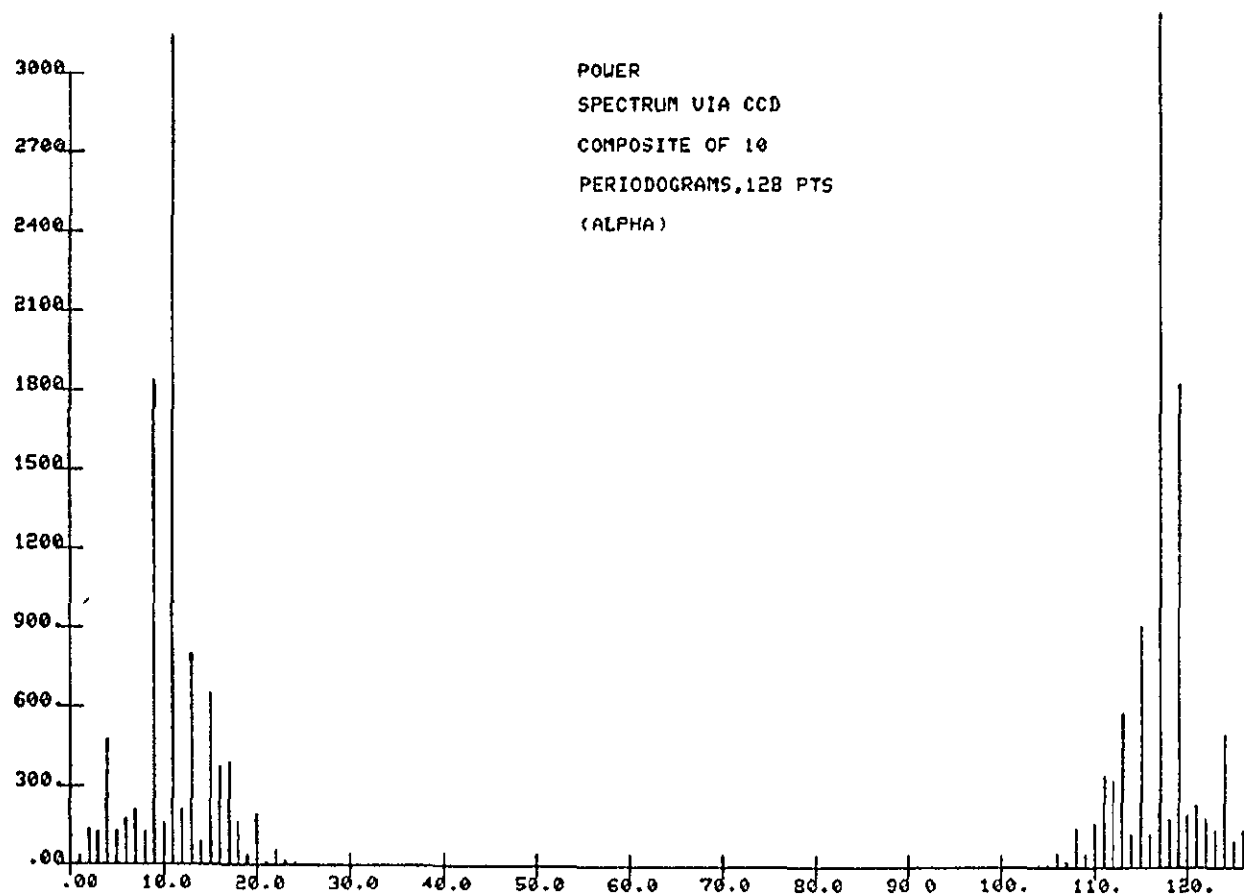


Figure 4-5. Power Spectrum Using CZT for the First
Averaged Periodogram for "ALPHA"

ORIGINAL PAGE IS
OF POOR QUALITY

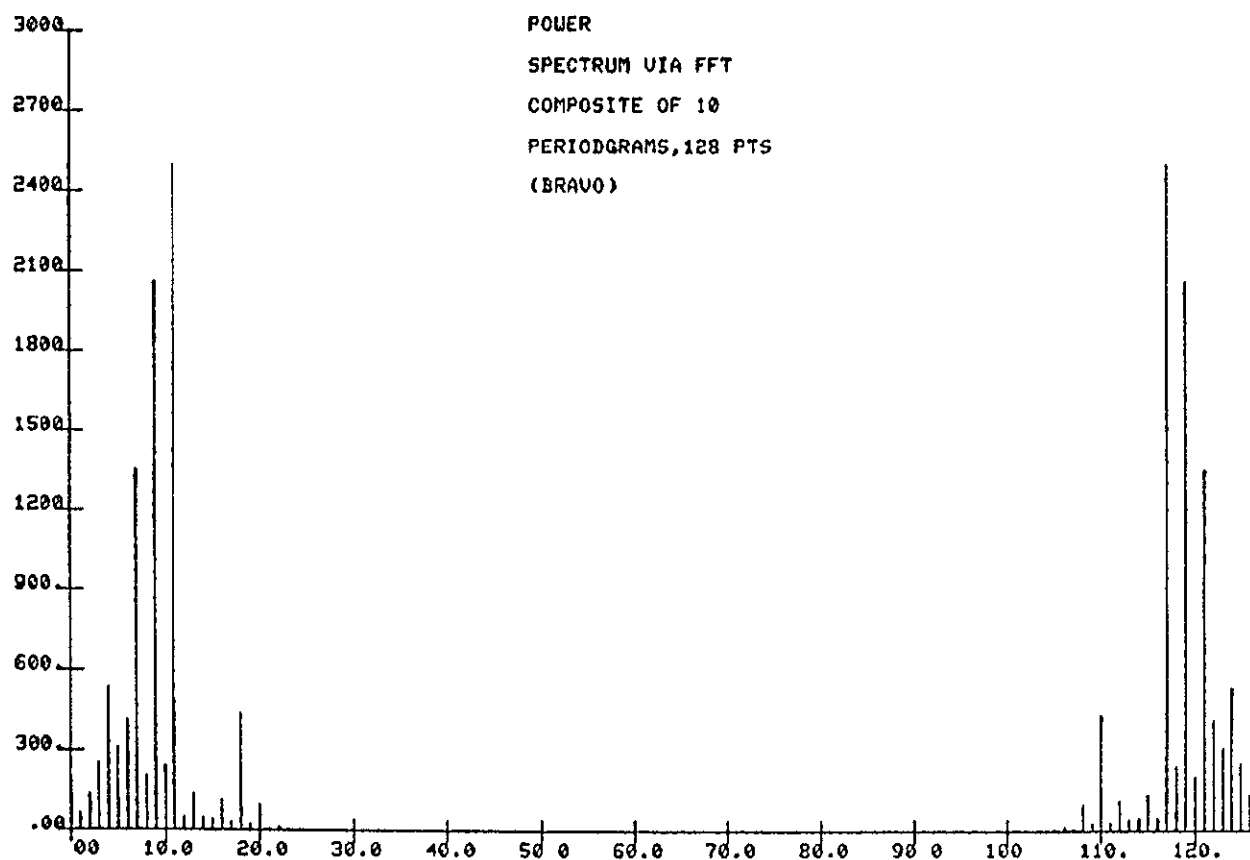


Figure 4-6. True Power Spectrum of the First Averaged
Periodogram of "Bravo"

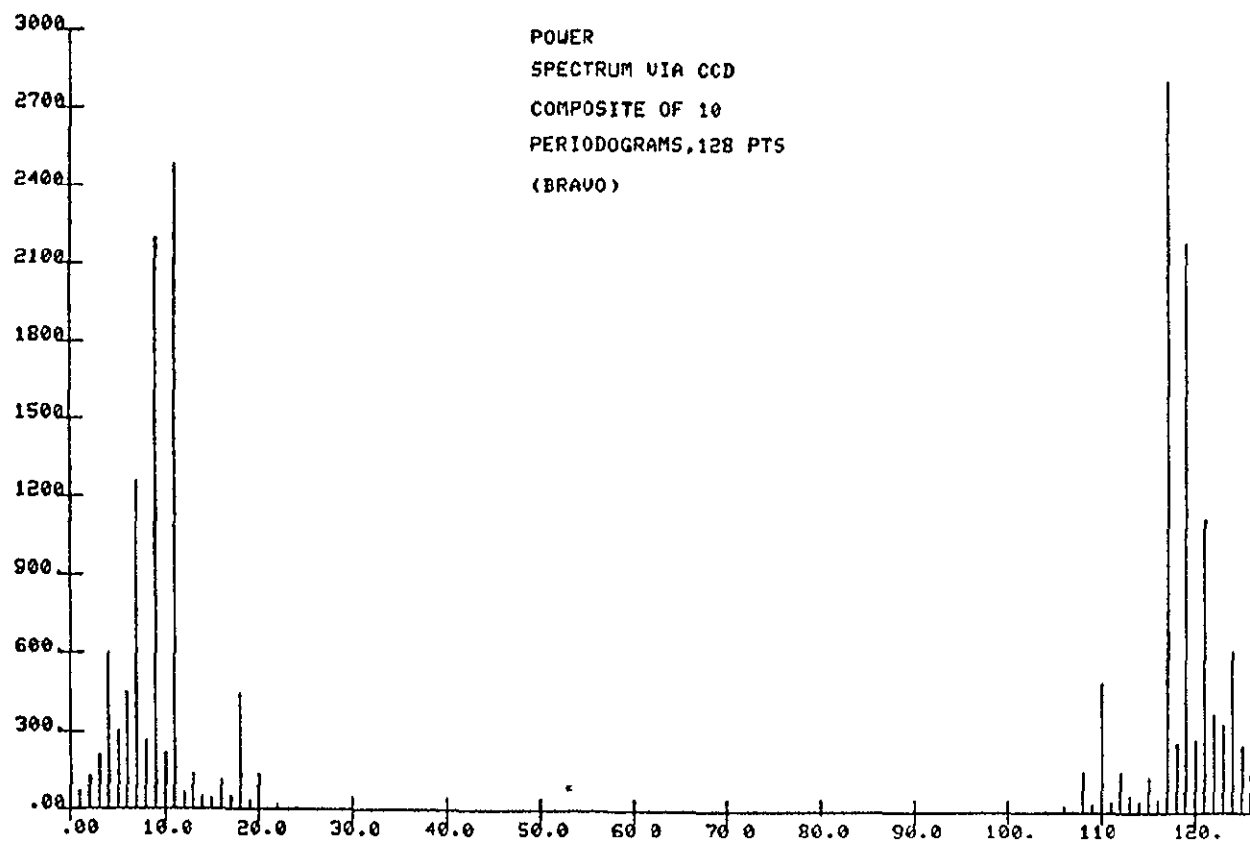


Figure 4-7. Power Spectrum via CZT for the First Averaged Periodogram of "Bravo"

APPENDIX IV-A

Charge-Coupled Device Simulation Program

```

FTN4      PROGRAM CCD
C
C      BIT 15 ON FOR INDIVIDUAL ,OFF FOR COMPOSITE
C
C      BIT 0 ON FOR EACH ITERATION RESULT
C
C      BIT 1 ON FOR LABEL
C
      DIMENSION SIN1(512),COS1(512),DATA1(512),DATA2(512)
      DIMENSION TABL(512),CMPOS(512)
      DIMENSION JUNK(10)
      DIMENSION NAME(40)
      COMMON IS1,IS2,IS3,IS4,IS5,IS6,R1,R2,R3,R4,IS7,IS8,R5,R6
      INTEGER A
      REAL ISPY
      WRITE(1,500)
500      FORMAT("INPUT TYPE OF PLOT -1.,0,1")
      READ(1,*)ITYP
      WRITE(1,501)
501      FORMAT("INPUT HI Y")
      READ(1,*)HIY
      WRITE(1,502)
502      FORMAT("TYPE 1 FOR LOG, 0 FOR LIN")
      READ(1,*)ILOG
      WRITE(1,2)
2      FORMAT("TYPE CCD WINDOW SIZE ")
      READ(1,*)N
      WRITE(1,400)
400      FORMAT("TYPE # WINDOWS FOR COMPOSITE")
      READ(1,*)NW
      CALL WINDO(0,1)
      WRITE(1,3)
3      FORMAT("TYPE TAPE FILE NUMBER")
      READ(1,*)IOP
      CALL PTAPE(8,IOP,0)
      WRITE(1,13)
13      FORMAT("TYPE ONE IF HEADER ON TAPE, ELSE 0")
      READ(1,*)IHD
      IF(IHD.EQ.0) GO TO 14
      DO 6 I=1,9
      READ(8,7)(JUNK(J),J=1,10)
7      FORMAT(10A2)
6      CONTINUE
      READ(8,8)(JUNK(J),J=1,10)
8      FORMAT(10I6)
14      CALL CONV(N,COS1,SIN1)
      DO 10 I=1,N
      READ(8)A
      DATA1(N-I+1)=A*COS1(I)/N
10      DATA2(N-I+1)=A*SIN1(I)/N
      DO 9 L=1,N

```

```

9      CMPOS(L)=0.
      LC=0
11     DO 120 L=1,NW
      LC=LC+1
      DO 100 I=1,N
      DO 70 I2=1,N-1
      IQ=N-I2
      DATA1(IQ+1)=DATA1(IQ)
70     DATA2(IQ+1)=DATA2(IQ)
      READ(8)A
      DATA1(I)=A*COS1(I)/N
      DATA2(I)=A*SIN1(I)/N
      CALL CONV(N,COS1,SIN1,DATA1,DATA2,RESL)
      I1=I-1
      TABL(I)=RESL
      CMPOS(I)=CMPOS(I)+(RESL-CMPOS(I))/LC
100    CONTINUE
      IF(ISSW(0))102,101
101    IF(L.NE.NW)GO TO 120
102    B1=N-1
      CALL CHRS(27,12,128)
      CALL RANGE(0.,B1,0.,HIY,0,ILOG)
      CALL AXES (0.,0.)
      ISPY=B1/10
      CALL TICK(0,0.,B1,ISPY,30,1)
      ISPY=HIY/10
      CALL TICK(1,0.,HIY,ISPY,30,1)
      IF(ISSW(15))115,116
115    CALL GRAPH(ITYP,N,TABL)
      GOTO 118
116    CALL GRAPH(ITYP,N,CMPOS)
118    IF (ISSW(1))119 , 120
119    WRITE(6,200)
200    FORMAT("SET CURSOR, STRIKE A KEY, AND INPUT LABEL:")
      CALL CURSI(IHD , IXPOS,IYPOS)
      CALL TPLOT(0,IXPOS,IYPOS)
      CALL CHRS(31,128,128)
      READ(6,210)(NAME(I),I=1,40)
210    FORMAT(40A1)
      WRITE(1,210)(NAME(I),I=1,40)
      WRITE(6,220)
220    FORMAT("TYPE 1 FOR MORE LABELS, ELSE 0")
      READ(6,*)IHD
      IF(IHD.EQ.1) GO TO 119
120    CONTINUE
      WRITE(6,12)
12     FORMAT("MORE DATA... 1 FOR YES ELSE 0 ")
      READ(6,*)ILP
      IF (ILP.EQ.1 )GOTO11
      END
C
C

```

```

SUBROUTINE CONV(N,COS1,SIN1,DATA1,DATA2,RESL)
DIMENSION COS1(1),SIN1(1),DATA1(1),DATA2(1)
DATAK/0/
K=K+1
IF(K.GT.1)GOTO 50
DO 10 I=1,N
  I1=I-1
  COS1(I)=COS(3.14159*I1*I1/N)
10  SIN1(I)=SIN(3.14159*I1*I1/N)
  RETURN
50  S1=0.
  C1=0.
  S2=0.
  C2=0.
  DO 70 I=1,N
    C1=C1+DATA1(I)*COS1(I)
    S1=S1+DATA1(I)*SIN1(I)
    S2=S2+DATA2(I)*SIN1(I)
70  C2=C2+DATA2(I)*COS1(I)
    RESL=(S2+C1)*(S2+C1)+(S1-C2)*(S1-C2)
  RETURN
END
END$

```

APPENDIX IV-B

Fast Fourier Transform Power Spectrum Program

```

FTN4
      PROGRAM CCFT
C
C
C      THIS PROGRAM COMPUTES THE POWER SPECTRUM OF N DIMENSIONAL
C      RECORDS OF DATA. A COMPOSITE SPECTRUM CAN ALSO BE GENERATED.
C
C      THIS PROGRAM IS THE ANALOG OF CCD PROGRAM.
C
C
C      BIT 15 ON FOR INDIVIDUAL ,OFF FOR COMPOSITE
C
C      BIT 0 ON FOR EACH ITERATION RESULT
C
C      BIT 1 ON FOR LABEL
C
      DIMENSION ARRY(2,512),TABL(512),CMPOS(512)
      DIMENSION JUNK(10),NAME(40)
      COMMON IS1,IS2,IS3,IS4,IS5,IS6,R1,R2,R3,R4,IS7,IS8,R5,R6
      INTEGER A
      REAL ISPY
      WRITE(1,500)
500    FORMAT("INPUT TYPE OF PLOT -1.,0,1")
      READ(1,*)ITYP
      WRITE(1,501)
501    FORMAT("INPUT HI Y")
      READ(1,*)HIY
      WRITE(1,502)
502    FORMAT("TYPE 1 FOR LOG, 0 FOR LIN")
      READ(1,*)ILOG
      WRITE(1,2)
2      FORMAT("TYPE FFT WINDOW SIZE")
      READ(1,*)N
      WRITE(1,400)
400    FORMAT("TYPE # WINDOWS FOR COMPOSITE")
      READ(1,*)NW
      CALL WINDO(0,1)
      WRITE(1,3)
3      FORMAT("TYPE TAPE FILE NUMBER")
      READ(1,*)IOP
      CALL PTAPE(8,IOP,0)
      WRITE(1,13)
13     FORMAT("TYPE ONE IF HEADER ON TAPE, ELSE 0")
      READ(1,*)IHD
      IF(IHD.EQ.0) GO TO 14
      DO 6 I=1,9
      READ(8,7)(JUNK(J),J=1,10)
7      FORMAT(10A2)
6      CONTINUE
      READ(8,8)(JUNK(J),J=1,10)
8      FORMAT(10I6)
14     DO 9 L=1,N

```

```

9      CMPOS(L)=0.
      LC=0
11     DO 120 L=1,NW
      LC=LC+1
      DO 50 I=1,N
      READ(8)A
      ARRAY(1,I)=FLOAT(A)/N
50     ARRAY(2,I)=0.0
      CALL FFT(ARRAY,N,-1)
      DO 55 I=1,N
      TABL(I)=ARRAY(1,I)*ARRAY(1,I)+ARRAY(2,I)*ARRAY(2,I)
55     CMPOS(I)=CMPOS(I)+(TABL(I)-CMPOS(I))/LC
      IF(ISSW(0))102,101
101    IF(L.NE.NW)GO TO 120
102    B1=N-1
      CALL CHRS(27,12,128)
      CALL RANGE(0.,B1,0.,HIY,0,ILOG)
      CALL AXES (0.,0.)
      ISPY=B1/10
      CALL TICK(0,0.,B1,ISPY,30,1)
      ISPY=HIY/10
      CALL TICK(1,0.,HIY,ISPY,30,1)
      IF(ISSW(15))115,116
115    CALL GRAPH(ITYP,N,TABL)
      GOTO 118
116    CALL GRAPH(ITYP,N,CMPOS)
118    IF (ISSW(1))119 , 120
119    WRITE(6,200)
200    FORMAT("SET CURSOR, STRIKE A KEY, AND INPUT LABEL:")
      CALL CURSI(IHD , IXPOS,IYPOS)
      CALL TPLOT(0,IXPOS,IYPOS)
      CALL CHRS(31,128,128)
      READ(6,210)(NAME(I),I=1,40)
210    FORMAT(40A1)
      WRITE(1,210)(NAME(I),I=1,40)
      WRITE(6,220)
220    FORMAT("TYPE 1 FOR MORE LABELS, ELSE 0")
      READ(6,*)IHD
      IF(IHD.EQ.1) GO TO 119
120    CONTINUE
      WRITE(6,12)
12     FORMAT("MORE DATA... 1 FOR YES ELSE 0 ")
      READ(6,*)ILP
      IF (ILP.EQ.1 )GOTO11
      END

C
C
      SUBROUTINE FFT(DATA,NN,ISIGN)
      DIMENSION DATA(1)
      SIGN=FLOAT(ISIGN)
      N=2*NN
      J=1

```

```

DO 5 I=1,N,2
IF(I-J)1,2,2
1  TEMPR=DATA(J)
   TEMPI=DATA(J+1)
   DATA(J)=DATA(I)
   DATA(J+1)=DATA(I+1)
   DATA(I)=TEMPR
   DATA(I+1)=TEMPI
2  M=N/2
3  IF(J-M)5,5,4
4  J=J-M
   M=M/2
   IF(M-2)5,3,3
5  J=J+M
   MMAX=2
6  IF(MMAX-N)7,9,9
7  ISTEP=2*MMAX
   ZAM=FLOAT(MMAX)
   DO 8 M=1,MMAX,2
   THETA=3.141592*FLOAT(M-1)/ZAM
   WR=COS(THETA)
   ARG=1.-WR*WR
   IF (ARG) 10,11,11
10  ARG=0.
11  WI=SIGN*SQRT(ARG)
   DO 8 I=M,N,ISTEP
   J=I+MMAX
   TEMPR=WR*DATA(J)-WI*DATA(J+1)
   TEMPI=WR*DATA(J+1)+WI*DATA(J)
   DATA(J)=DATA(I)-TEMPR
   DATA(J+1)=DATA(I+1)-TEMPI
   DATA(I)=DATA(I)+TEMPR
8  DATA(I+1)=DATA(I+1)+TEMPI
   MMAX=ISTEP
   GO TO 6
9  RETURN
END
END$

```

BIBLIOGRAPHY

1. Schafer, Ronald W., Rabiner, Lawrence R., "Parametric Representation of Speech," in D. Raj Reddy (Ed.), Speech Recognition, Academic Press, Inc., New York, 1975, pp. 99-150.
2. Ichikawa, Akira, Nakano, Yasuaki, Nakata, Kazuo, "Evaluation of Various Parameter Sets in Spoken Digits Evaluation," IEEE Transactions on Audio and Electroacoustics, Vol. AU-21, No. 3 (June, 1973), pp. 202-209.
3. Hatan, Jean-Paul, "A Practical Application of a Real-Time Isolated Word Recognition System Using Syntactic Constraints," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-22, No. 6 (December, 1974), pp. 416-419.
4. Benz, H. F., private communications, NASA, Langley, Va., 1975.
5. Rabiner, L. R. and Schafer, R. W., "The Chirp z-Transform," IEEE Transactions on Audio and Electroacoustics, Vol. AU-17, No. 2, June, 1969, pp. 86-92.
6. McDonald, W. E., Jr., "Use of the ORS Feature Set for Automatic Speech Recognition," Masters Thesis, University of Virginia, August 1976.

V. RESOLUTION ENHANCEMENT

Introduction

A novel application of charge-coupled devices (CCDs) for data processing to a unique electro-optical system has been studied and is presented in this part of the report. The electro-optical system employs a technique for binary monochromatic resolution enhancement beyond the capability of an individual sensor. The system utilizes a gray-level to binary-level conversion and sensor motion to extrapolate data values smaller than the individual sensor area resolution capability.

Such a system may find practical application in such diverse fields as satellite and aerial reconnaissance to medical tomography or any application where greater resolving power is needed than a discrete sensor can provide. Since the technique requires only a sensor capable of gray-level information, any further advances in technology should only serve to further the technique's utility. With recent studies of multi-level logic systems [1] and with CCDs inherent capability for multi-level output, a suggested projection would be a combination of sensor and processing system which could contribute to a "smart sensor."

Pursuing such a promising technique into realization, the implementation, however, poses some difficulty due to the characteristically large amounts of data generated. This data, when processed by McRee's technique [2], becomes bulky. A paper by Henderson and McVey [3], discussed later, proposes a more sophisticated algorithm which alleviates some of the conditions imposed by McRee at the expense of even more data manipulation and storage. A mode of reducing such manipulation or immense storage, yet not at the expense of the improved Henderson technique's benefits, is deemed necessary. Such a mode of reducing the manipulations or storage might exist in the form of a filtration process occurring at some location in the system. With data available in either sequential or block information, particularly if a square matrix or picture sensory array is used, a two-dimensional filter or a high-speed sequential filter would be indicated. With high speed, low power/density dissipation, and analog and digital processing, CCDs appear to be ideal for this application. This

is reinforced by their flexibility in assuming the form of either a two-dimensional or high-speed sequential filter.

This application shall be approached in this report by first presenting and exploring McRee's technique [2] into the detailed phase of an example. Next, Henderson's algorithm [3] shall be presented in a similar manner with the last part of the presentation being devoted to the practical application of CCD technology. The practical application phase is the major topic presented and will be pursued in depth as various segments of the research are developed.

The presentation of the developed research shall progress through five phases. Phase one introduces previous research effort by McRee, et al. In phase two, several possible CCD applications for the enhanced resolution system are developed with phase three consisting of the phase two applications presented as examples to illustrate the limitations of each application. Sources of error in the system algorithm and in the device implementation will be discussed in the fourth phase. The last phase concludes the report with a summary, recommendations, and conclusions based on the illustrative examples.

In the method for enhancing resolution in electro-optical systems with a multi-level output that has been presented by McRee and McVey [2], the resolution enhancement entails special logic circuits which allow discrimination of events smaller than the area of the discrete sensor. In a spatial transformation of numeric systems, gray-level information of a fixed resolution is mapped onto a binary-level system with a resulting resolution greater than that of the gray-level system. McRee has formulated a linear transformation equation

$$S = MT \tag{5-1}$$

for this conversion in addition to a technique for its realization, where S is defined as a vector composed of the elements s_{rc} which are the sensor's output with r as row and c as column; M is defined as a $p \times q$ transformation matrix with m_{ij} elements and $p < q$ with p sensors and q outputs; and T is

defined as a vector composed of the elements t_{ij} , where i is row and j is column in the target area.

McRee's technique typically utilizes a solid-state, electro-optical sensor array with an individual sensor area A and side length d , where $A = d^2$. Refer to Fig. 5-1. In the examples, a five discrete level output sensor with a resolution improvement of 4:1 is assumed.

Positioning a gridwork over a target area and establishing a direction of travel, as depicted in Fig. 5-2, the array is first positioned as in Fig. 5-3. Samples of the target are processed, whereby the amount of light energy incident on the sensor is integrated; and this integrated input is passed to the special processing logic circuitry. The processed output, if exceeding 50% of the sensor's maximum output, is converted to a 1 state, if otherwise, to a \emptyset state, i.e.,

$$t_{ij} = \begin{cases} 1, & \text{if 50\% sensor area is black} \\ \emptyset, & \text{if otherwise} \end{cases} \quad (5-2)$$

This value is stored, the sensor moved to the second position (see Fig. 5-3), and a signed algebraic addition is performed between this data and the previously stored value. This result is stored and the process continued until the target has been completely sampled. The matrix of binary data thus formed is the enhanced resolution output.

McRee found that the enhanced resolution matrix was $p \times q$, where $p < q$ thus incurring a non-square matrix and the resultant inversion difficulties. To overcome this imposition McRee established a boundary condition (refer to Fig. 5-4), where

$$t_{1j} = t_{i1} = 1 \quad (5-3)$$

which reduces his M matrix to a square and invertible matrix.

The only other apparent restriction is that there is a tendency for error to accumulate and propagate as is illustrated in the following theory

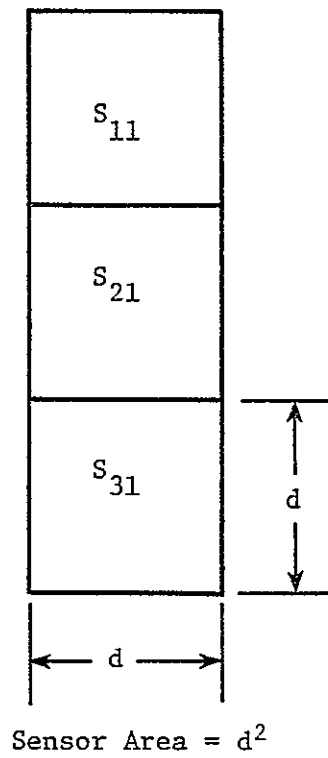


Figure 5-1. Sensor Area of Individual Sensor on Array

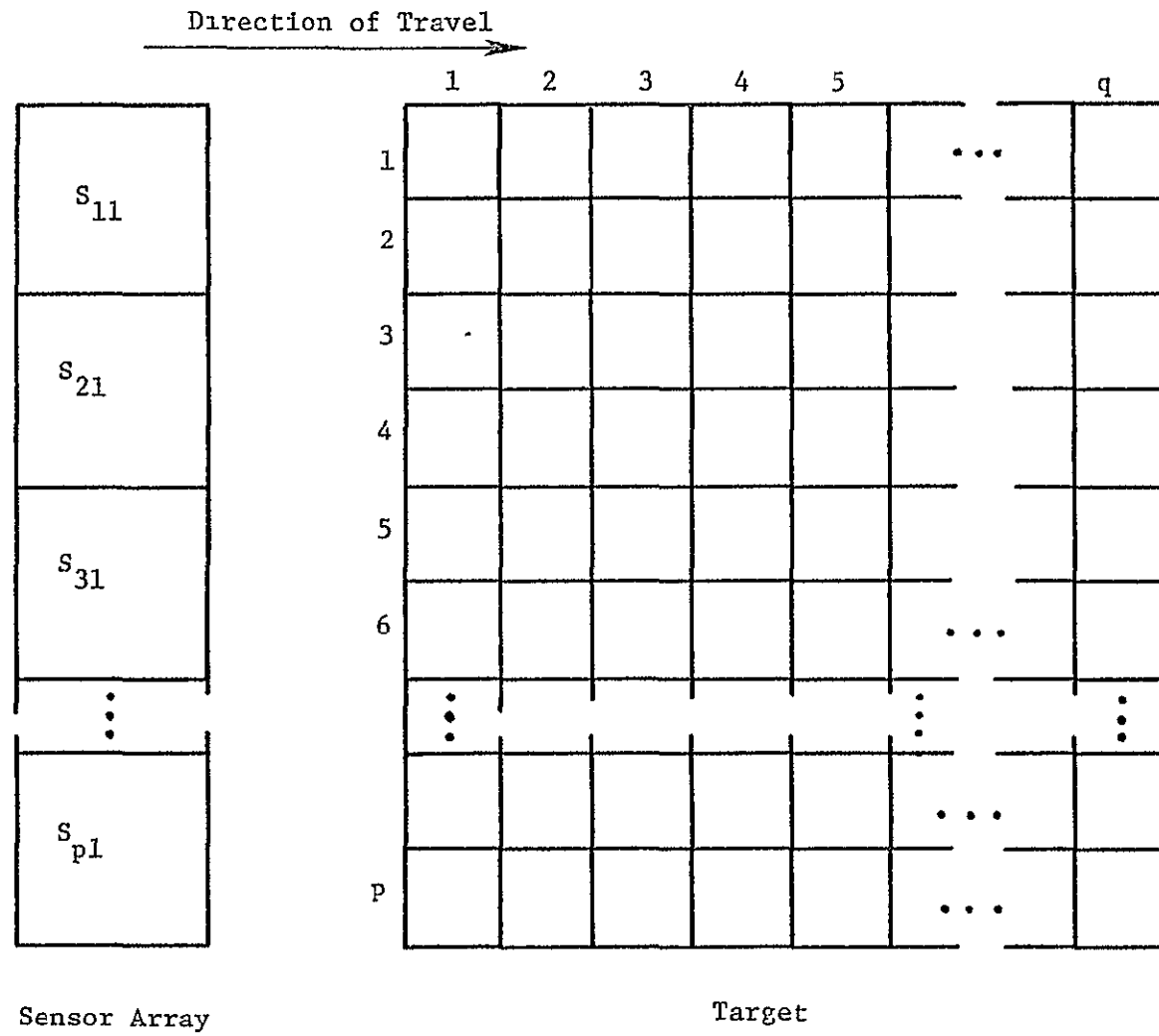


Figure 5-2. Sensor Array's Direction of Travel and Illustration of Target Grid

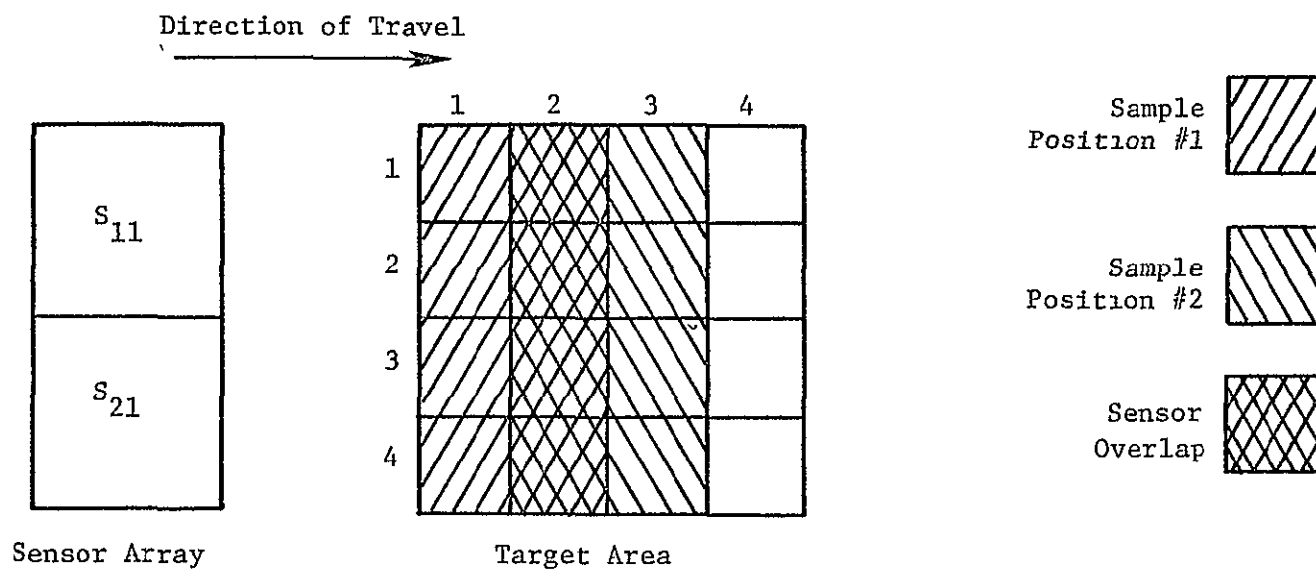


Figure 5-3 . Sensor Array Motion for Two Sample Positions and Resulting Sensor Array Overlap

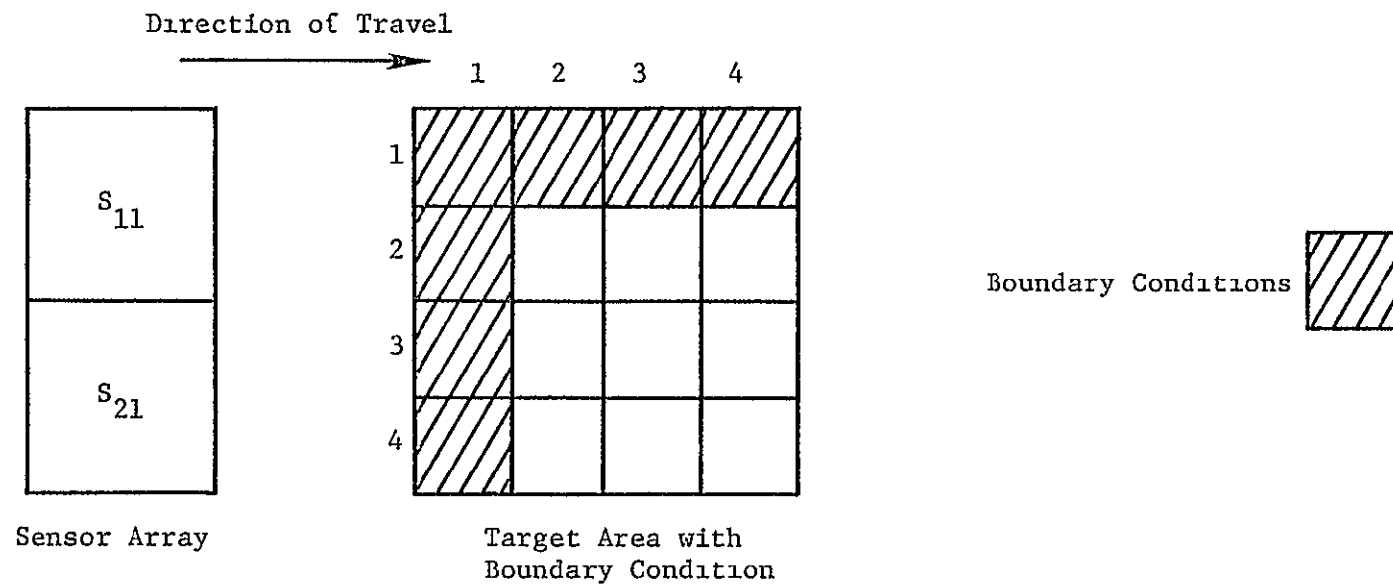


Figure 5-4 . Sensor Array and Target Area with Boundary Condition Established

and example. These restrictions are overcome by Henderson's application of estimation techniques to the data which shall be illustrated.

McRee defines the process of transforming gray information to enhanced binary information as image delimitation. As is obvious from Fig. 5-3, the enhanced binary resolution is obtained from sensory overlap and the resultant increase in image threshold determination accuracy. This accuracy appears from the signed algebraic addition performed by the special logic circuits to ascertain the exact location of the sensor loci which, in effect, increases resolution.

A perfect sensor is the first assumption by McRee in his restrictions upon the technique justified by several techniques for the elimination of dead space between sensors. That sensor area does not approach zero is another assumption which is based on practicality. The last two assumptions cause the most difficulty, as shall be discussed. The resultant non-square matrix problem is overcome by establishing boundary conditions; and, by assuming an error-free or noiseless transformation commencing at the boundary, McRee introduces a major source of error.

For the transformation matrix M and its subsequent unit vector columns \bar{m}_j , let there exist a coefficient l_j such that

$$l_1 \bar{m}_1 + l_2 \bar{m}_2 + \dots + l_q \bar{m}_q = \bar{0} \text{ for } j = (1, 2, 3, \dots, q) \quad (5-4)$$

as a test for linear dependence and has a set $l_j = \{-1, 0, +1\}$. However, McRee sets any value $l_j \leq 0$ equal to zero, thereby reducing the set to $l_j = \{0, 1\}$. Then, the linear transformation equation may be represented by an example.

Arranging a sensor array, as in McRee's paper and in Fig. 5-5, it must be pointed out that there must exist a complete overlap, either through exaggerated sensor motion or staggered sensors, for the results to be meaningful. For an example target image, as in Fig. 5-6, the following manipulations are performed from (1), $S = MT$,

$$S_{rc}(t) = \sum_{j=1}^q m_{ij} t_{1j} \quad (5-5)$$

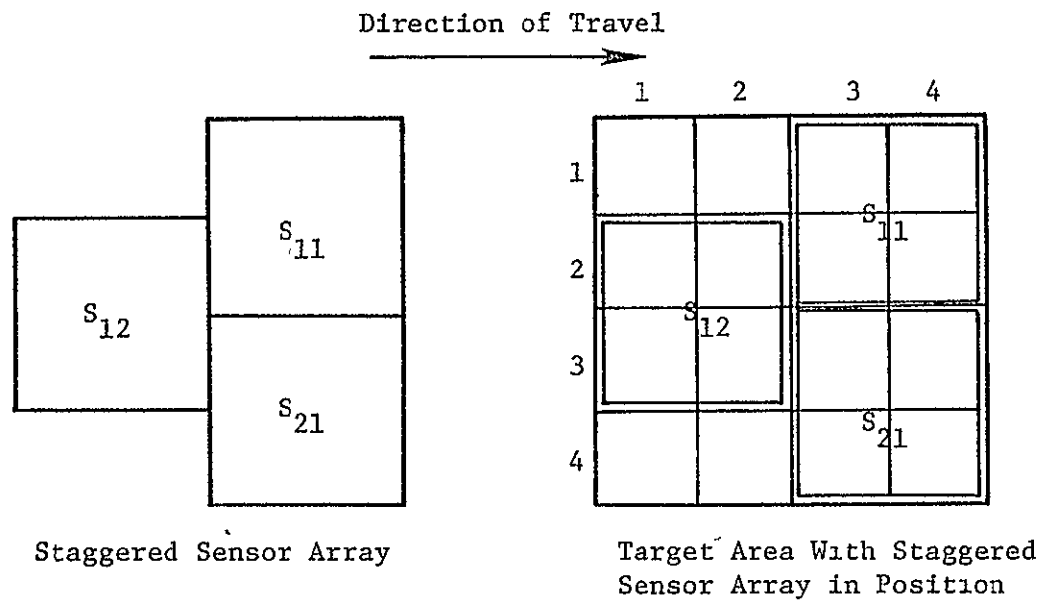


Figure 5-5. Staggered Sensor Array Shown Positioned Over Target Area

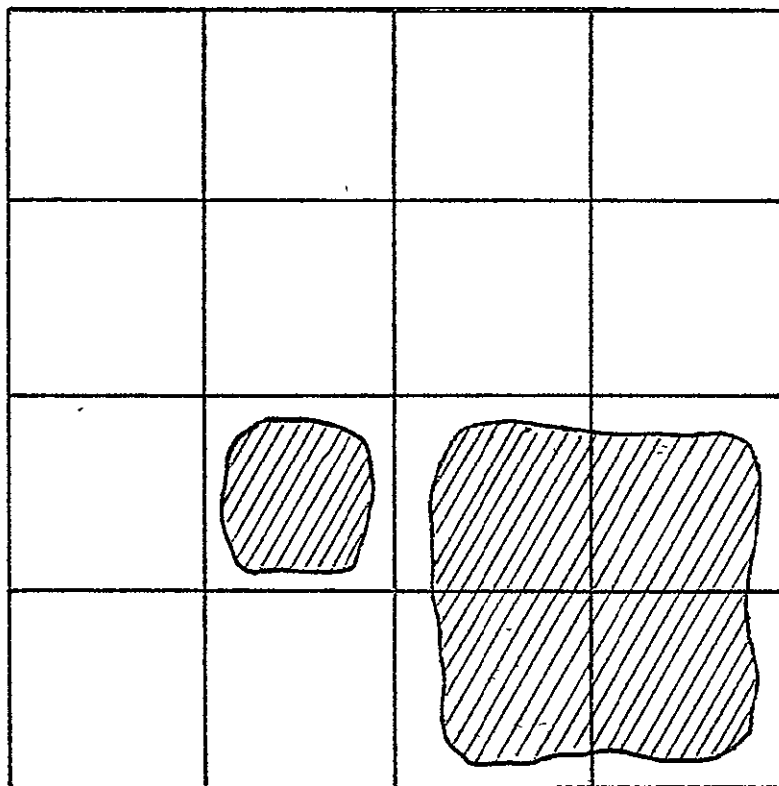


Figure 5-6. Target Area with Sample Target.

setting t as time for sensor positions, as in Fig. 5-7, and n as the number of columns in the sensor to be processed, it is seen that $p = nt + (n - 1)t = (2n - 1)t$ and $q = 2nt$ for the transformation matrix M . The equation

$$\begin{bmatrix} s_{11}(1) \\ s_{11}(2) \\ s_{11}(3) \\ s_{12}(1) \\ s_{12}(2) \\ s_{12}(3) \\ s_{21}(1) \\ s_{21}(2) \\ s_{21}(3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} t_{11} \\ t_{12} \\ t_{13} \\ t_{14} \\ t_{21} \\ t_{22} \\ t_{23} \\ t_{24} \\ t_{31} \\ t_{32} \\ t_{33} \\ t_{34} \\ t_{41} \\ t_{42} \\ t_{43} \\ t_{44} \end{bmatrix} \quad (5-6)$$

results. However, if boundary conditions, as in (5-4), are implemented, the equation above reduces to that of (5-7), where $p = q = (2n - 1)t$, as illustrated in Fig. 5-8 and

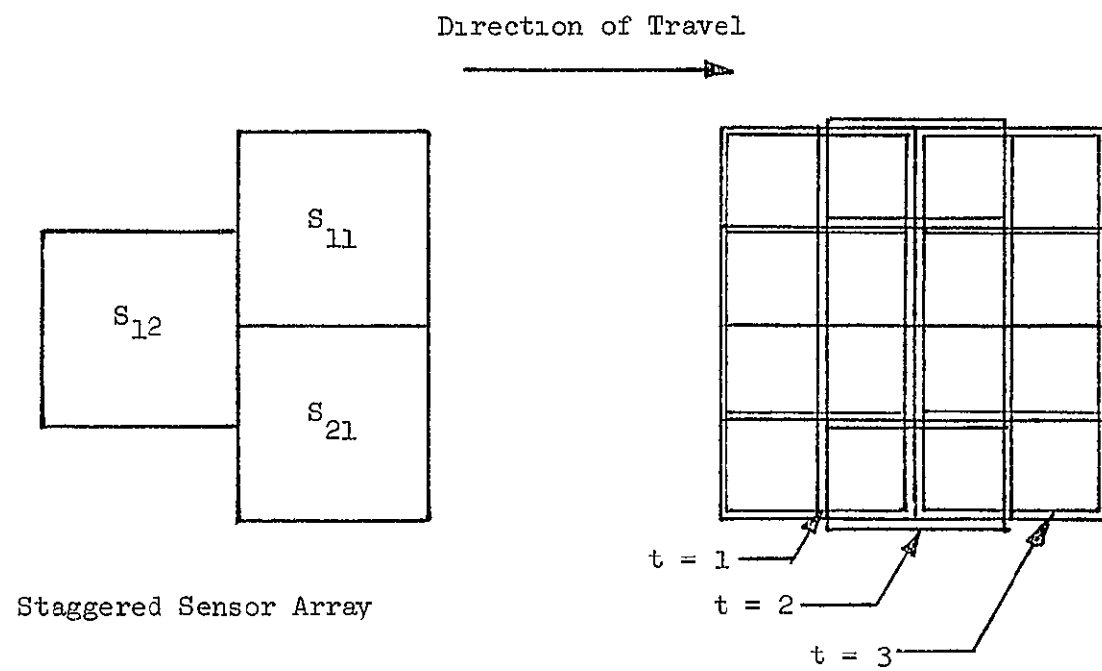


Figure 5-7. Staggered Sensor Array Positions Shown for Each Discrete Sample Time Position.

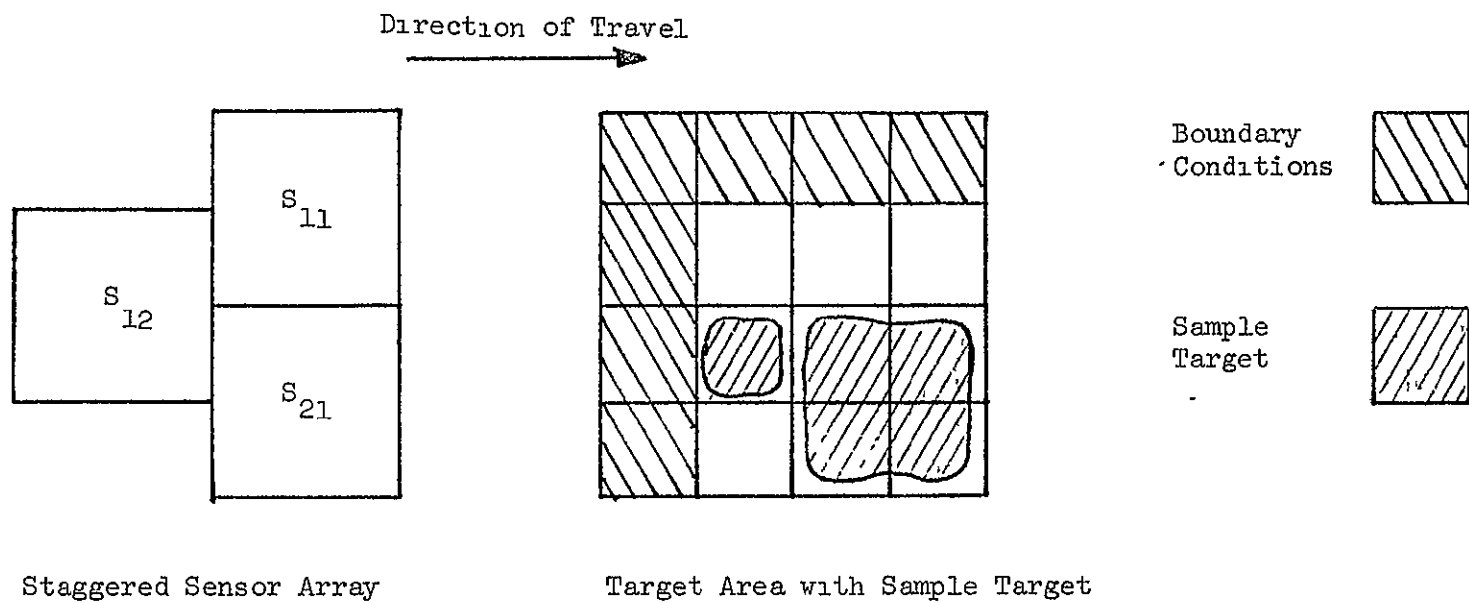


Figure 5-8. Staggered Sensor Array with Sample Target Illustrating Boundary Conditions

$$\begin{bmatrix} s_{11}(1) - 3 \\ s_{11}(2) - 2 \\ s_{11}(3) - 2 \\ s_{12}(1) - 2 \\ s_{12}(2) \\ s_{12}(3) \\ s_{21}(1) - 2 \\ s_{21}(2) \\ s_{21}(3) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} t_{22} \\ t_{23} \\ t_{24} \\ t_{32} \\ t_{33} \\ t_{34} \\ t_{42} \\ t_{43} \\ t_{44} \end{bmatrix} \quad (5-7)$$

Inverting and performing the matrix multiplication, the equation above can be expressed as its components which are the enhanced binary resolution algebraic equations.

$$\begin{bmatrix} t_{22} \\ t_{23} \\ t_{24} \\ t_{32} \\ t_{33} \\ t_{34} \\ t_{42} \\ t_{43} \\ t_{44} \end{bmatrix} = \begin{bmatrix} s_{11}(1) - 3 \\ s_{11}(2) - s_{11}(1) + 1 \\ s_{11}(3) - s_{11}(2) + s_{11}(1) - 3 \\ s_{12}(1) - s_{11}(1) + 1 \\ s_{12}(2) - s_{12}(1) - s_{11}(2) + s_{11}(1) + 1 \\ s_{12}(3) - s_{12}(2) + s_{12}(1) - s_{11}(3) + s_{11}(2) - s_{11}(1) + 1 \\ s_{21}(1) - s_{12}(1) + s_{11}(1) - 3 \\ s_{21}(2) - s_{21}(1) - s_{12}(2) + s_{12}(1) + s_{11}(2) - s_{11}(1) + 1 \\ s_{21}(3) - s_{21}(2) + s_{21}(1) - s_{12}(3) + s_{12}(2) - s_{12}(1) + s_{11}(3) - s_{11}(2) + s_{11}(1) - 3 \end{bmatrix} \quad (5-8)$$

Then, substituting values for the above

$$s_{rc}(t) = [3 \ 2 \ 2 \ 3 \ 2 \ 2 \ 3 \ 3 \ 4]^T \quad (15-9a)$$

$$T = [0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 1]^T \quad (5-9b)$$

vectors result concluding the example (refer to Fig. 5-9).

McRee has made two assumptions which give rise to difficulties as they are generalized. With the first assumption noted before that gives difficulty, McRee assumes a boundary condition which reduces the transformation matrix M to a square matrix; however, such a boundary condition interferes with the practical implementation. The second troublesome assumption, that the transformation is error-free or noiseless, is seen from the algebraic equations previously described to propagate error in the recursive technique. Henderson proposes and justifies the removal of the imposing boundary conditions as well as ridding the model of the propagating error by application of estimation techniques to generate a pseudoinverse. The purpose of Henderson's technique was to eliminate the boundary conditions; however, a side benefit not discussed was the removal of the propagated and accumulated error.

Directing attention to the removal of these restrictions, Henderson begins with a presentation of the classical estimation model

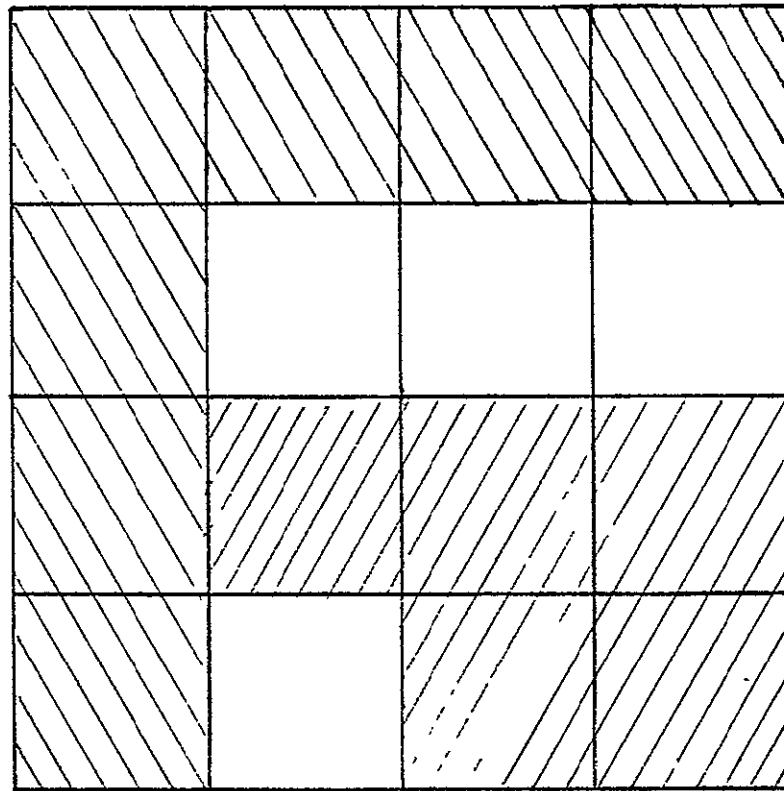
$$z = Hx + u \quad (5-10)$$

where, $z = m \times 1$ observation vector, $H = m \times n$ mapping matrix, $x = n \times 1$ parameter vector, and $u = m \times 1$ vector of random variables or noise. By the direct substitution of McRee's notation into the estimation equation (5-1) and (5-11), the only changes are the resultant treatment of the inverse transformation matrix

$$S = MT + u \quad (5-11)$$

Henderson treats the estimate as noise-free and applies least squares to obtain

$$\hat{T} = M^+ S \quad (5-12)$$



Boundary
Conditions



Sample
Target



Figure 5-9. Enhanced Resolution Sample Target Transformation of McRee's Technique from Sample in Figure 5-8.

where M^+ is the pseudoinverse matrix of M , and \hat{T} is the estimate vector. Utilizing the maximum row rank property of M , the pseudoinverse is given by

$$M^+ = M^T (MM^T)^{-1} \quad (5-13)$$

where the M^T matrix is not a complex matrix by prior definition. From the previous target example for McRee's technique, the sensor values are assigned (5-9a) and the pseudoinverse computations performed by TSO BASIC on a HP2100 time-share system resulting in

$$T = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1]^T. \quad (5-14)$$

Several target transformations are illustrated in Figs. 5-10, 5-11, and 5-12. The M^+ matrix is computed only once, and (5-14) can be summarized by (5-15), where the number of coefficients is determined by the number of sensors and the resolution enhancement ratio, where k , the threshold of the image delimitation, is typically set equal to .5 and

$$t_{ij} = K \sum_{i=1}^{pt} m_{ij}^+ s_{rc}(t) \quad (5-15)$$

This equation is implemented as the design equation for the special processing circuitry and is suggestive of the transversal filter design equation (5-16) given by [4]

$$V_d(t) = \sum_{m=1}^k h_m d(t - mw) \quad (5-16)$$

where w is the tap delay, k is the number of taps, h_m is the weighting coefficient, t is the sample interval, d is the input sample voltage, and $V_d(t)$ is the filtered output voltage.

The combination of the techniques of McRee and Henderson has progressed to the stage of a practical implementation. The implementation from the equation (5-15), as conceived by McRee, involves addition and multiplication as well as an analog-to-digital conversion for each sensor as per the flow

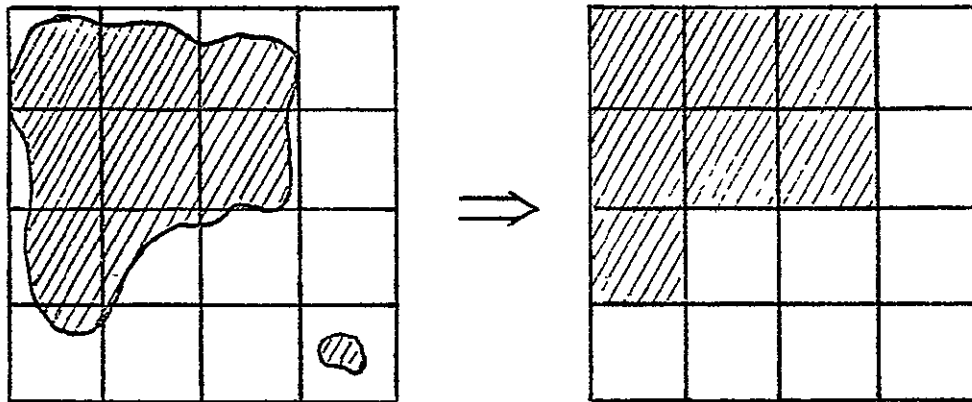


Figure 5-10. Transformation Example of Henderson's Algorithm.

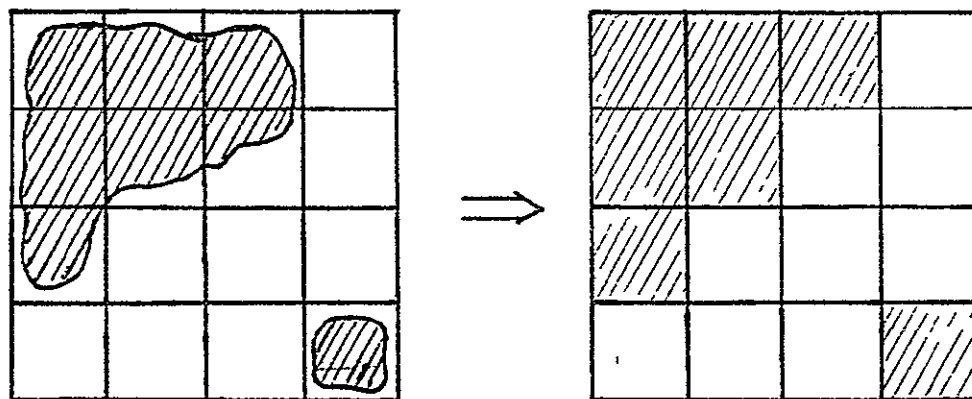


Figure 5-11. Transformation Example of Henderson's Algorithm.

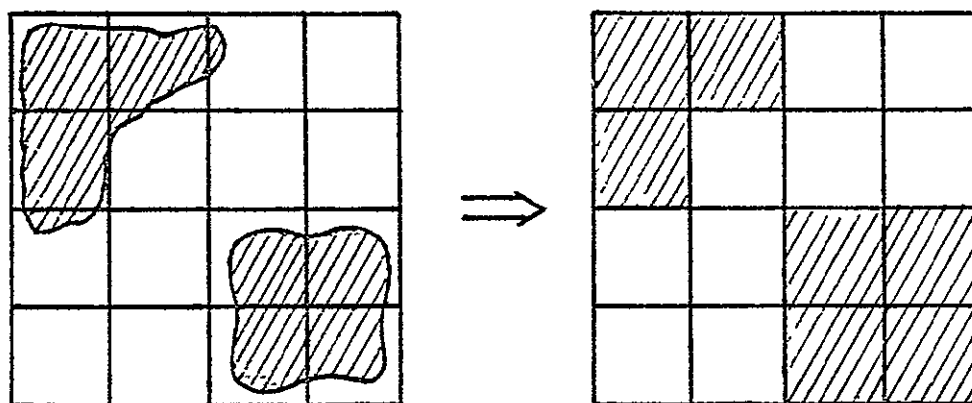


Figure 5-12. Transformation Example of Henderson's Algorithm.

chart process summary, Fig. 5-13. However, if analog information can be processed instead of digital, a great savings in size and complexity could be achieved as well as to reduce the error inherent in quantization. In the transversal filter equation (5-16), with its continuous data sampling nature, this achievement is apparent. Therefore, the design equation (5-15) implemented as a transversal filter would be an obvious realization; and, with CCDs, several design realizations of transversal filters [4, 5, 6, and 7] have been accomplished.

System Designs †

Transversal filters will be used in the implementation. A general form of the transversal filter is shown in Fig. 5-14 and is described by

$$V_d(t) = \sum_{m=1}^k h_m d(t - mw) \quad (5-16)$$

as given previously. This configuration uses a normalized electrode tapped a proportional amount across the width corresponding to the ratio [8]

$$(1 + h_m)/(1 - h_m) \quad (5-17)$$

where h_m is a coefficient of range $-1 < h_m < +1$. The ratio results in the difference amplifier output [8] of

$$\begin{aligned} v_{out} &\propto \sum_{m=1}^k 1/2(1 + h_m)Q_m - \sum_{m=1}^k 1/2(1 - h_m)Q_m \\ &= \sum_{m=1}^k h_m Q_m \end{aligned} \quad (5-18)$$

where h_m is the weighting coefficient, and Q_m is the input charge sample in the cell. From (5-16), the function $d(t - mw)$ implies a time delay or a sample storage. The term transversal filter is applied to this general class of device, particularly where a storage array is implied and the data shifted into the register is not part of a continuous input. Consequently, the classical definition of the use of the filter as a signal detector or emphasis device shall not apply. Instead, discrete sensor

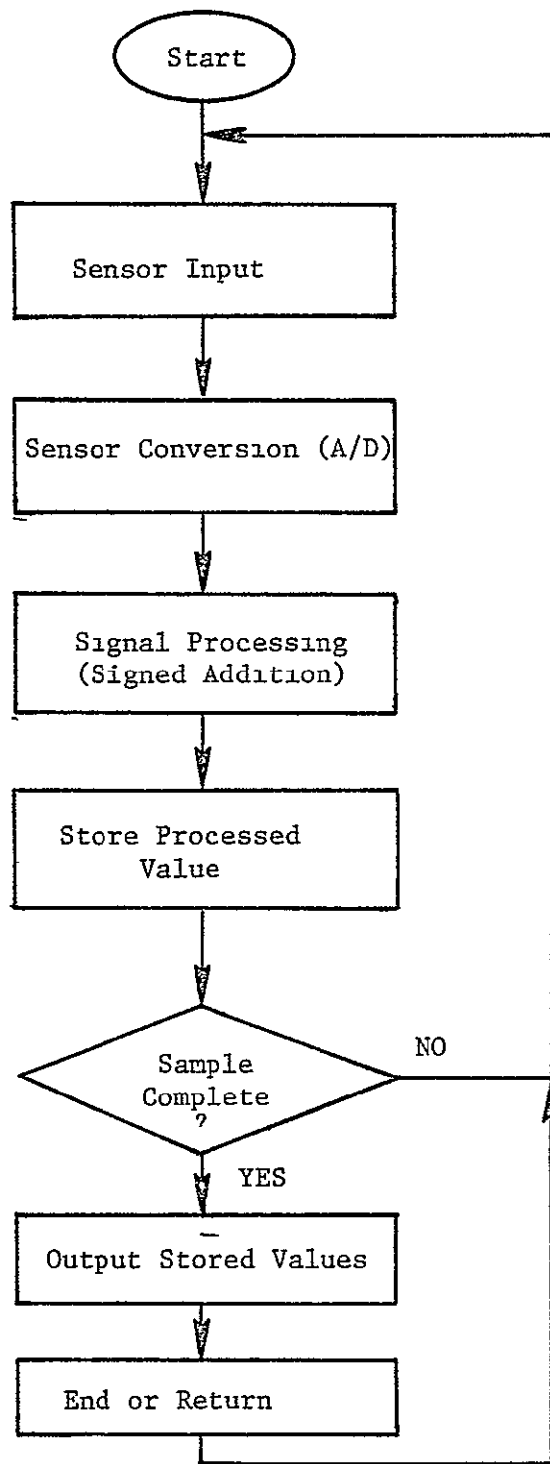


Figure 5-13. Simplified Flow Chart of McRee's Technique

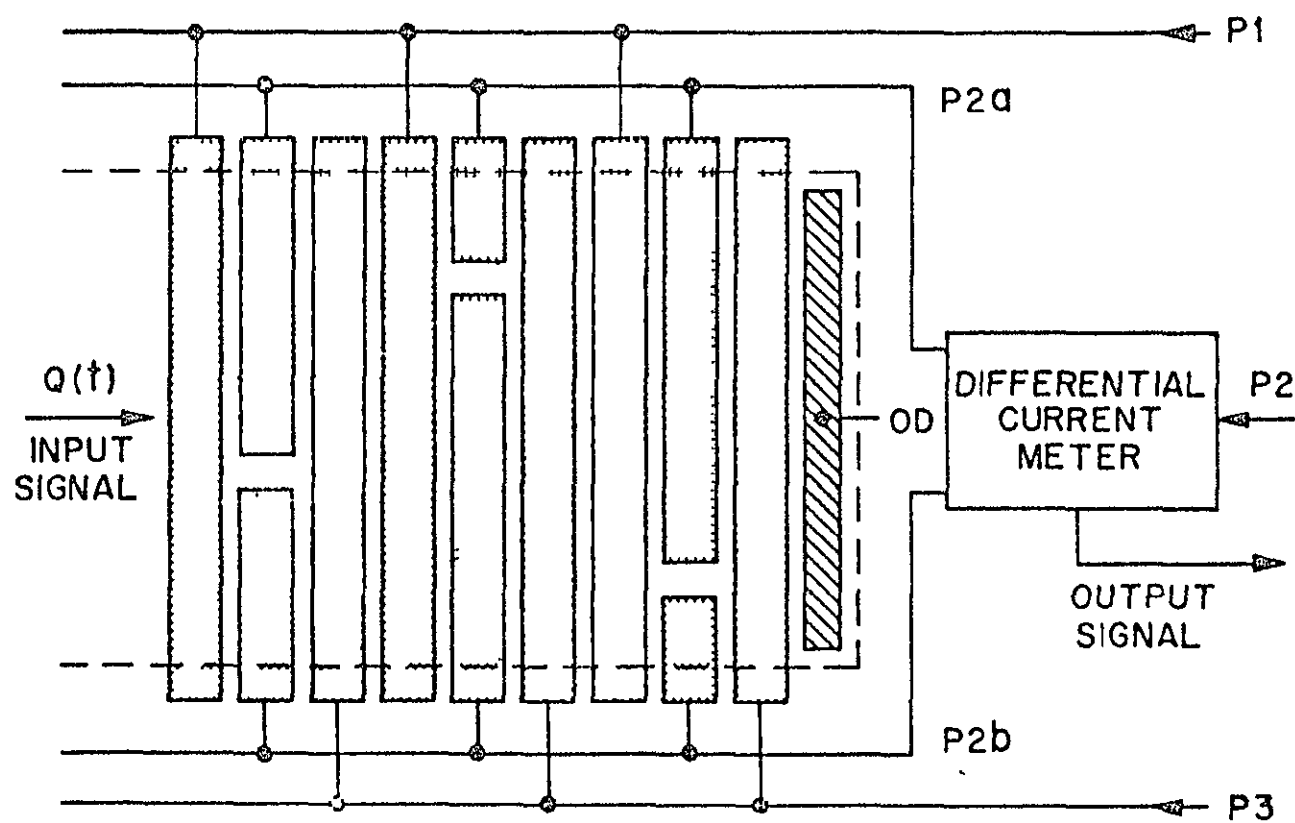


Figure 5-14. Typical Weighted Tap Transversal Filter Arrangement [4].

values processed in a matrix format where each sensor output for a sample is weighted by a coefficient, then summed with the remaining weighted sensor values to form an element output, shall be termed a transversal filter or just filter.

To increase the flexibility of the transversal filter, adjustable tap weights implemented in place of the fixed tap weights would realize a programmable filter. Several techniques for realizing a programmable transversal filter have been implemented [4, 6, 9, 10]. As mentioned in Section III, none are available as off-the-shelf components.

Considering the most direct method or a minimal design and maximal hardware, there are two immediate approaches to a direct implementation of which one would be the application of one transversal filter to each output element as shown in Fig. 5-15. Such a design would employ a maximal number of filters; and, as seen from the figures, there are many modes of data input and output. Utilizing a maximal filter design would result in the highest throughput of data, particularly if parallel input/output structures are adopted; however, the expense of such a high data rate is increased circuit density. It becomes apparent from minimal size limitations in CCDs and the consequential trade-offs between accuracy, error, and size that the ensuing number of filters would increase with the dimensionality of the pseudoinverse M^+ . Since, in the future, there will undoubtedly be an increase in technological capability and the demand for higher speed processing shall continue, the maximal filter design will be used to introduce the design examples and discussion.

With the processing of information in an analog form, an immediate savings in a conversion unit (A/D) per sensor, as proposed by McRee, is realized; whereas, if CCD technology is employed in the sensor and processing circuitry, a direct interface is then possible. A one-to-one correspondence between the filter and resolution element is best fulfilled with fixed tap weights in a transversal filter where each coefficient represents a column element in a row or the pseudoinverse M^+ . The transversal filter, as described by (5-16), then sums the weighted elements which are shifted along the weighting delay line until all samples are

From Sensor Array

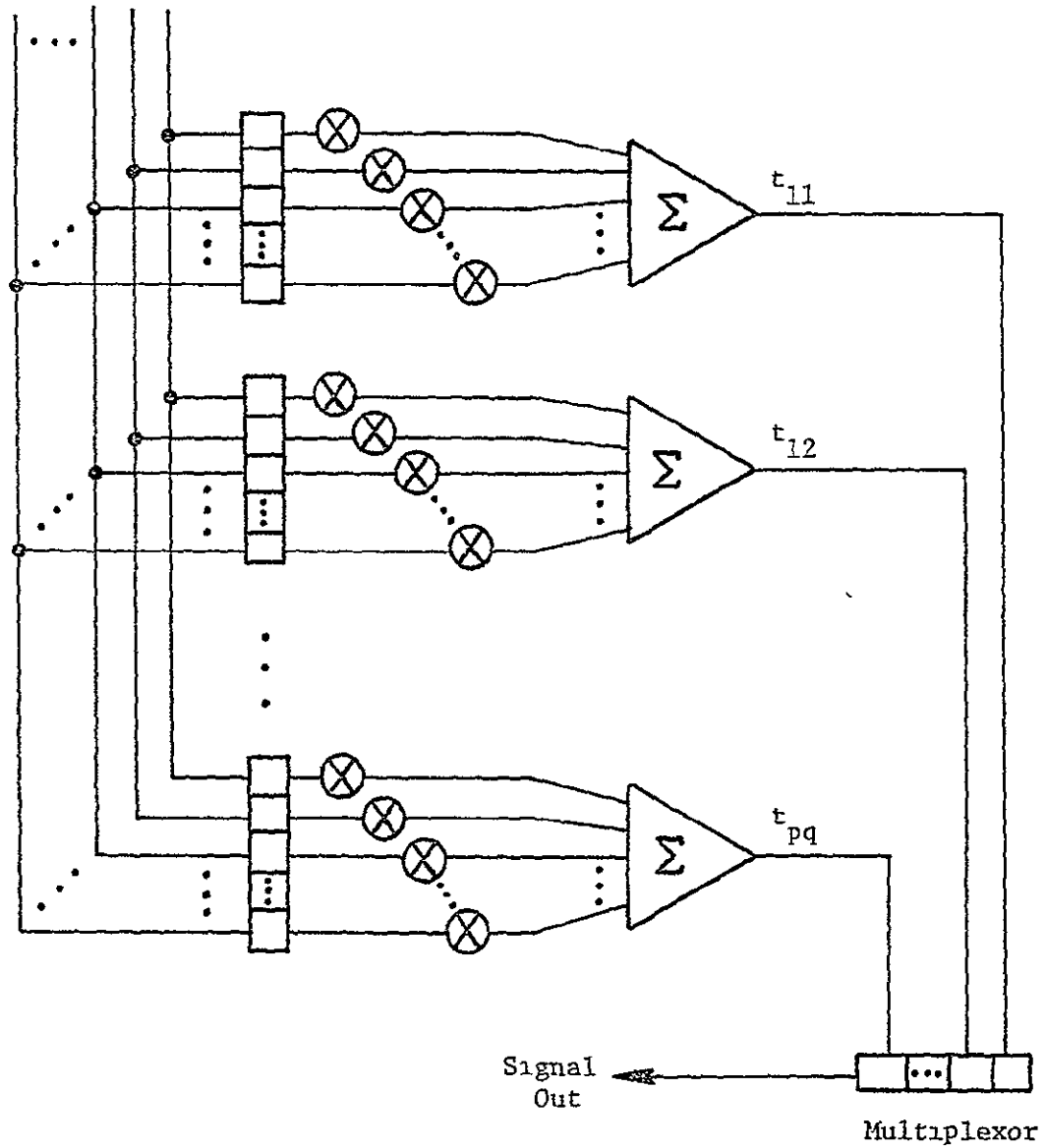


Figure 5-15. System Design Utilizing a Transversal Filter per Output Element

present in correct order. The output is strobed at the proper instant to either load a multiplexor for a serial output which would ease the conversion to a standard TV transmission pattern or load parallel output buffers. A bonus evident in the output circuitry is that no additional clock speed is required in the serial multiplexor scheme, since there is a finite time period between successive conversions through the filter, as illustrated by the timing diagram in Fig. 5-16. The timing diagram can be expressed by

$$\text{COMPLETE SAMPLE CONVERSION TIME} = T_c = \frac{d}{f_s} \frac{f_{co}}{f_{cs}} = \frac{d}{f_s} \quad (5-19)$$

where f_{cs} is the shift clock frequency for the filter, f_s is the sensor sample rate, d is the sensor resolution enhancement factor, and f_{co} is the output clock frequency.

Two input schemes are indicated by the serial shifting nature of the transversal filter. One intuitive scheme would be a simple serial shift into the transversal filter, as depicted in Fig. 5-17; the other scheme would involve a parallel-to-serial conversion with the effect of increasing the speed of the data transfer given the higher clock speed of the filter. Another scheme in the same vein of parallel loading might be realized if the filter is amiable in a density trade-off, as indicated in Fig. 5-18, to a loading in the filter area. Thus, having seen the maximal filter approach, an attempt at the maximal design and minimal hardware approach will be presented as another implementation.

Such an approach would require a single transversal filter with a variable programmable tap weighting, as described by several experimental research groups [4, 6, 9, 10]. The same input and output schemes could be utilized with the output possessing an additional intermediate analog delay line storage to store the sums-of-products terms from each sensor sample position until the complete sample is obtained. The pseudoinverse M^+ matrix must be loaded into the programmable filter a row at a time. If row-by-row programming is to be the mode of operation, an analog output storage is present, and the filter shift register readout is nondestructive and can be recirculated, then no additional input circuitry is necessary.

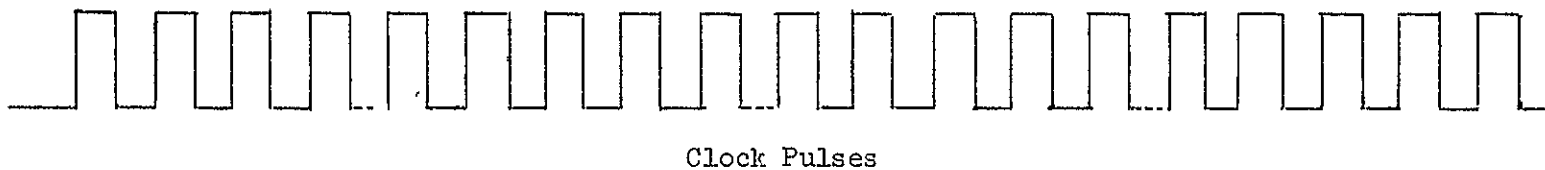
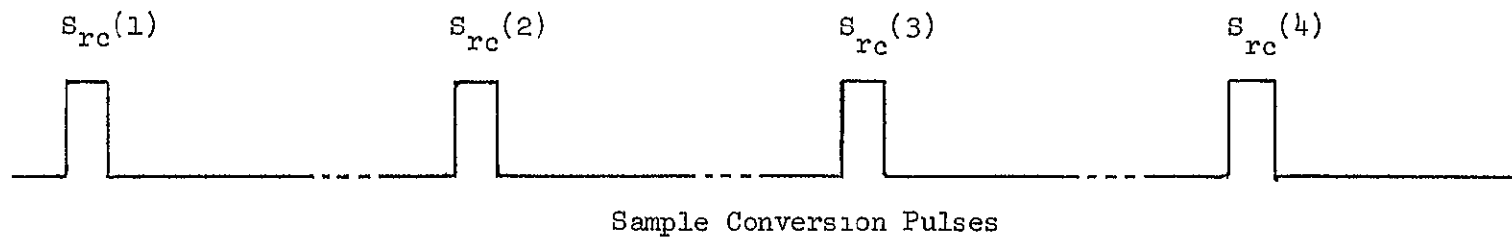


Figure 5-16. Timing Diagram Illustrating the Finite Time Available Between Sample Conversions.

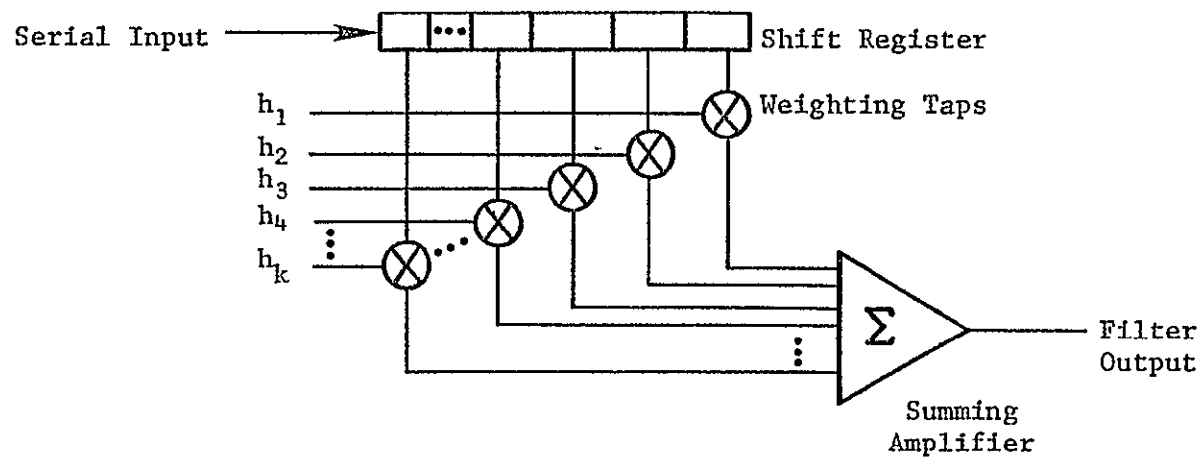


Figure 5-17. Serial Loading of a Transversal Filter

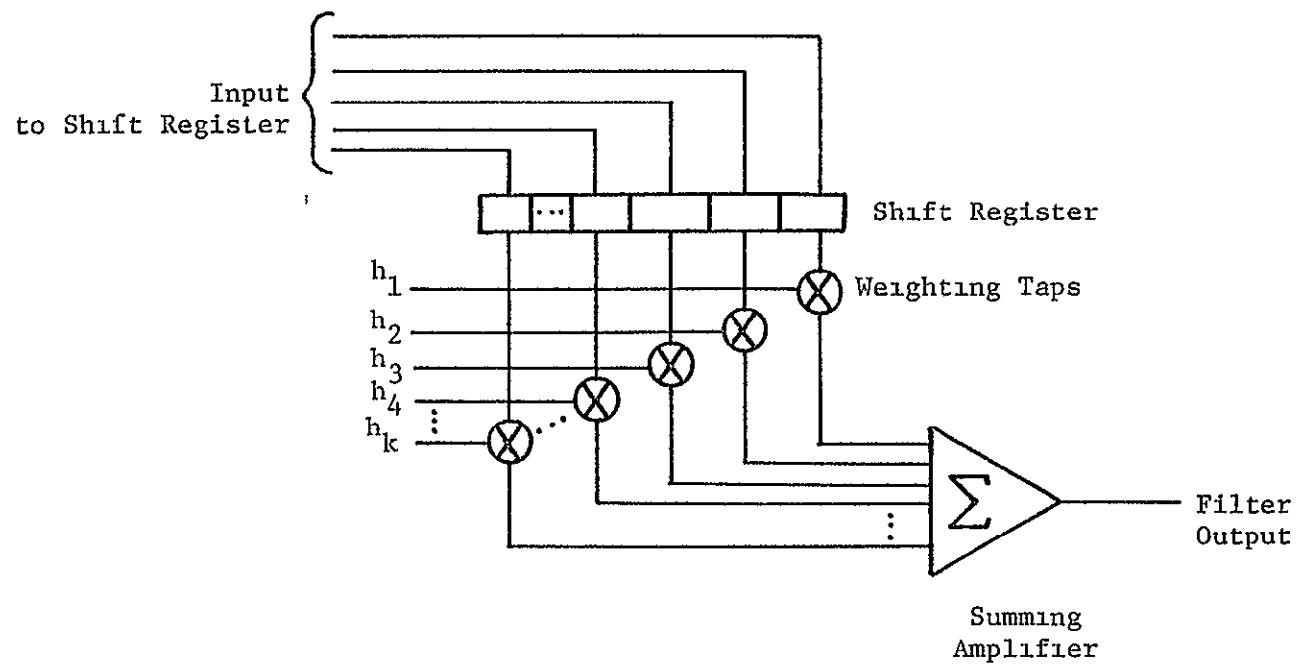


Figure 5-18. Parallel Loading of a Transversal Filter

However, if the filter cannot be recirculated or if analog output storage is not present, then there must exist an input storage of sufficient size to retain a complete set of sensor readings corresponding to

$$S_{in} = n \times d \quad (5-20)$$

where n is the number of sensors and d is the resolution enhancement factor (e.g., 4:1). There still exists the need of storage for the coefficient M^+ matrix whose dimensionality has been previously discussed and examples presented. It is apparent that a larger amount of discrete storage cells would be necessary if signed digital storage were to be used instead of an analog value, owing to the greater radix values available. Digital storage technology is well established and continues to increase in density; however, there are only two basic types of analog storage currently available. The first type is evident in the analog delay line and is comparable to a dynamic storage unit or to a recirculating shift register. This dynamic storage requires a refresh scheme which could pose problems in its numbers, particularly, as a large number of sensors are employed; but, if externally accessible, there is the alternative of an adaptive filtering scheme. The second type is seen as a static type employing a MOSFET structure in conjunction with a charge storage cell [4, 9] or as a tap weight where the tap area determines the value for a normalized charge which is then merged into the data stream in the shift register (Fig. 5-19). If the MOSFET/capacitor structure is employed and a reset line is available, then there also exists the alternative of an adaptive filter scheme. With storage of either digital or analog values, there still is the imposing dimensionality of the pseudo-inverse M^+ matrix, as evidenced by a small sensor matrix of 20×20 elements requiring 144,400 coefficients to be stored. Another example of a 100×100 element sensor matrix requires 9.8×10^7 coefficients which would serve to illustrate the exceeding large storage necessary for the coefficient matrix. There appears to be no easy implementation for the resolution enhancement technique; therefore, a further study of the coefficient matrix is deemed advisable.

Indeed, by studying the pseudoinverse M^+ matrix, a pattern of values appears to emerge and even the number of discrete values in the matrix seems

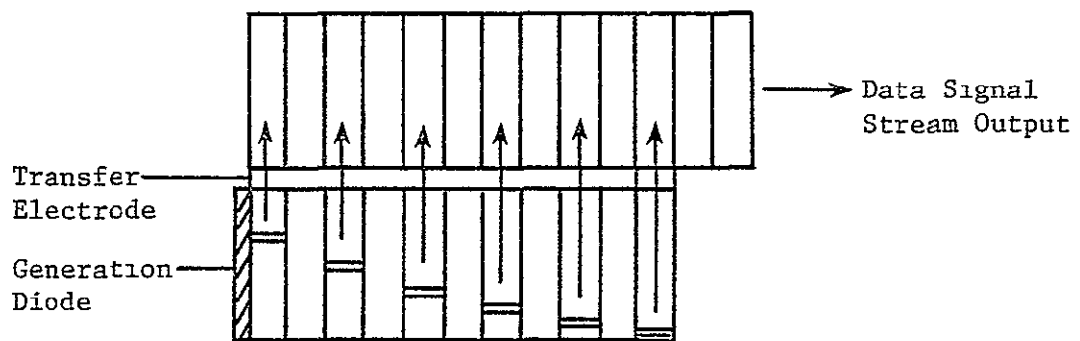


Figure 5-19. Example of Simple Analog Storage Using a Proportional Weighted Tap

limited. In constructing the transformation matrix (5-21), a pattern of diagonalized square unit submatrices is evident and such a pattern would intuitively give rise to a comparable pattern in the pseudoinverse M^+ matrix. A pattern does appear in the form of a mirror image reflected about a contral axis. Also, after several examples of various sensor array sizes were checked a consistent rule of thumb for the number of discrete absolute values for the coefficients is intuitively seen to be

$$V = 2n \quad (5-22)$$

where n is the number of sensors. An extensive study of the pattern displayed in the pseudoinverse reveals further possible reduction by considering the submatrices defined by the boundaries in (5-23). If each sensor array sample is considered to be a major submatrix of n columns where n is the number of sensors, then, minor submatrices composed of redundant elements can be formed. Redundant rows can be coded as to the number of iterations in the minor submatrix by the addition of a sufficient number of digital bits. Such a technique could be called microcoding and, to maintain consistency, shall be so termed. An instant reduction of the dimensionality of the pseudoinverse to that of a microcoded coefficient array brings increased concentration on the pattern in the pseudoinverse. There even appears to be a mirror image in the coefficient values in the major submatrices, thereby offering a further reduction in dimensionality by continuing to microcode or by a hardware design. If microcoding were to be continued, then the storage space would have to be doubled for each addition bit, whereas, if a hardware controller were added with a recirculating control register containing the pattern sequence code, the storage array would not increase. Such hardware control sequence code shall be termed program microcoding. There may be a combination of the two techniques that reduces the pseudoinverse dimensionality to the intuitively determined minimum of twice the number of sensors.

Beginning with the $2n$ coefficient array storage necessary for the basic system, a maximal control microcoding would maintain the minimal $2n$ storage area, whereas any additional coding in the form of program

Equation (5-21)

List of Values

$$A = .694444$$
$$B = .555556$$

C = .416667

$$D = .277778$$

E = .138889

$$F = .444444$$
$$G = .333333$$
$$H = .222222$$

I = .111111

$$J = .25$$
$$K = .166667$$
$$L = .083333$$
$$M = .027780$$
$$N = .055556$$
$$M^+ =$$
[illegible]

NOTE: Values displayed are absolute. Signs form redundant patterns in the pseudoinverse and can be programmed by the control circuitry with a single bit for a sign.

Equation (5-23)

microcoding would serve to swell the storage area. However, a type of storage for the controlled loading of the programmable filter must be provided and this constitutes an increased storage area even though this appears to be of a minimal nature due to the pattern redundances and the independence from the coefficient array. For the moment, further discussion of microcoding will be suspended in favor of a brief look at a storage arrangement of the coefficient values in an attempt to clarify the reasoning for the bias in the resultant choice of a microcoding technique.

Recalling again the pseudoinverse M^+ matrix and the major submatrices indicated in Eq. (5-23), an apparent technique would treat the rows composing the major submatrix as words to be loaded into an intermediate register preparatory to programming the filter. Since there appears to be two different words per minor submatrix, two program or address counters and appropriately gating logic (decoding) to the intermediate register would be indicated by the unit step increment change between word blocks in adjacent minor submatrices. Thus, a storage arrangement, as described, would maintain minimal $2n$ storage, yet ease the microcoding requirement by reducing the number of control sequences needed to replicate the pseudoinverse M^+ matrix.

Returning to the microcoding discussion, because the storage arrangement has simplified the controlling requirements only the control microcoding will be discussed. Examining the minor submatrices in the first major submatrix sample, a suggested axis of rotation would be approximately half the rows as depicted by the double lines in Fig. 5-20. A hardware design might incorporate a control counter to count to half the number of rows to generate an overflow which is routed to logic that converts the program or address counter to a down or reversed sequence counter thereby reversing the order of row display. The overflow pulse or charge packet also reroutes steering logic in the decoder to reverse the order of the columns thus implementing a mirror image of half of the major submatrix. A control counter may be designed as a ring counter or racetrack shift register driven by clock pulses. Such a counter is easily realized in CCds [11]. Having presented a technique for implementing the mirror image seemingly present in each major submatrix and a mode of reconstructing the

M^+

												Major Submatrix
A E E E E E E	E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	
B I I I I I I	B I I I I I I	D N N N N N N	D N N N N N N	D N N N N N N	D N N N N N N	D N N N N N N	D N N N N N N	D N N N N N N	D N N N N N N	D N N N N N N	D N N N N N N	
C L L L L L L	C L L L L L L	C L L L L L L	C L L L L L L	C L L L L L L	C L L L L L L	C L L L L L L	C L L L L L L	C L L L L L L	C L L L L L L	C L L L L L L	C L L L L L L	
D N N N N N N	D N N N N N N	D N N N N N N	D N N N N N N	D N N N N N N	D N N N N N N	B I I I I I I	B I I I I I I	B I I I I I I	B I I I I I I	B I I I I I I	B I I I I I I	
E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	E M M M M M M	A E E E E E E	A E E E E E E	A E E E E E E	A E E E E E E	
B B D D D D D	I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	
F F H H H H H	F F H H H H H	H H I I I I I	H H I I I I I	H H I I I I I	H H I I I I I	H H I I I I I	H H I I I I I	H H I I I I I	H H I I I I I	H H I I I I I	H H I I I I I	
G G K K K K K	G G K K K K K	G G K K K K K	G G K K K K K	G G K K K K K	G G K K K K K	G G K K K K K	G G K K K K K	G G K K K K K	G G K K K K K	G G K K K K K	G G K K K K K	
H H I I I I I	H H I I I I I	H H I I I I I	H H I I I I I	H H I I I I I	H H I I I I I	F F H H H H H	F F H H H H H	F F H H H H H	F F H H H H H	F F H H H H H	F F H H H H H	
I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	I I N N N N N	B B D D D D D	B B D D D D D	B B D D D D D	B B D D D D D	
C C C C C C C	L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	
G G G G G G G	G G G G G G G	K K K K K K K	K K K K K K K	K K K K K K K	K K K K K K K	K K K K K K K	K K K K K K K	K K K K K K K	K K K K K K K	K K K K K K K	K K K K K K K	
J J J J J J J	J J J J J J J	J J J J J J J	J J J J J J J	J J J J J J J	J J J J J J J	J J J J J J J	J J J J J J J	J J J J J J J	J J J J J J J	J J J J J J J	J J J J J J J	
K K K K K K K	K K K K K K K	K K K K K K K	K K K K K K K	K K K K K K K	K K K K K K K	G G G G G G G	G G G G G G G	G G G G G G G	G G G G G G G	G G G G G G G	G G G G G G G	
L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	L L L L L L L	C C C C C C C	C C C C C C C	C C C C C C C	C C C C C C C	
D D D D B B B	N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	
H H H H F F F	H H H H F F F	I I I I H H H	I I I I H H H	I I I I H H H	I I I I H H H	I I I I H H H	I I I I H H H	I I I I H H H	I I I I H H H	I I I I H H H	I I I I H H H	
K K K K G G G	K K K K G G G	K K K K G G G	K K K K G G G	K K K K G G G	K K K K G G G	K K K K G G G	K K K K G G G	K K K K G G G	K K K K G G G	K K K K G G G	K K K K G G G	
I I I I H H H	I I I I H H H	I I I I H H H	I I I I H H H	I I I I H H H	I I I I H H H	H H H H F F F	H H H H F F F	H H H H F F F	H H H H F F F	H H H H F F F	H H H H F F F	
N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	N N N N I I I	D D D D B B B	D D D D B B B	D D D D B B B	D D D D B B B	
E E E E E A	M M M M M E	M M M M M E	M M M M M E	M M M M M E	M M M M M E	M M M M M E	M M M M M E	M M M M M E	M M M M M E	M M M M M E	M M M M M E	
I I I I I B	I I I I I B	N N N N N D	N N N N N D	N N N N N D	N N N N N D	N N N N N D	N N N N N D	N N N N N D	N N N N N D	N N N N N D	N N N N N D	
L L L L L C	L L L L L C	L L L L L C	L L L L L C	L L L L L C	L L L L L C	L L L L L C	L L L L L C	L L L L L C	L L L L L C	L L L L L C	L L L L L C	
N N N N N D	N N N N N D	N N N N N D	N N N N N D	N N N N N D	N N N N N D	I I I I I B	I I I I I B	I I I I I B	I I I I I B	I I I I I B	I I I I I B	
M M M M M E	M M M M M E	M M M M M E	M M M M M E	M M M M M E	M M M M M E	M M M M M E	M M M M M E	E E E E E A	E E E E E A	E E E E E A	E E E E E A	
												Minor Submatrix

Figure 5-20. Pseudoinverse Matrix Illustrating Mirror Images in Major Submatrices.

rows for each sample major submatrix, the entire scheme could be depicted as in Fig. 5-22. It can be seen that the last remaining block to be described is the decoder of which several methods can be utilized to the greatest advantage. One such method would utilize a multiplexor to selectively load a serial shift register in a coded sequence; another method would employ CCD register routing by a multiplexor in much the same manner as before, except that the loading would be done in a parallel fashion (Fig. 5-23).

A greater reduction in size might be achieved if the programmable filter were deleted. Instead of shifting a normalized unit charge and parallel loading the output with the proportional analog value (Fig. 20), the sensor outputs are parallel loaded into the analog storage device. The tapped weighting electrodes are floating-gate amplifiers connected to the proper sign on the differential current meter. This realization would use B storage devices of the length of n sensor cells, where B is defined as the sum of the number of distinct row values per major submatrix over the entire completed sample. Decoding and control circuitry would be implemented similar to that described for the programmable filter resulting in a greater savings in complexity (Fig. 5-24).

Most attention has been directed at the circuitry for a single major submatrix sample. Structures which incorporate the total coefficient matrix will now be considered. A direct technique would attempt to process all the sensor output samples through a single transversal filter, reducing the number of filters but increasing the processing time with a small addition to the circuitry. Another technique has been suggested [7] which employs a staged approach easily seen if each major submatrix is thought of as a stage thereby separating the samples into a natural division. The Westinghouse report [7] confines itself to the direct storage of each individual element in a coefficient array, and as previously seen, this would be an impractical array. Next, an example design will be given to test for an approximate maximum number of sensors that could be fabricated on a single chip containing the signal processing circuitry.

In an attempt to generalize the design, fixed tap weight shift cells and normal linear shift cell shall have a constant area A . The programmable

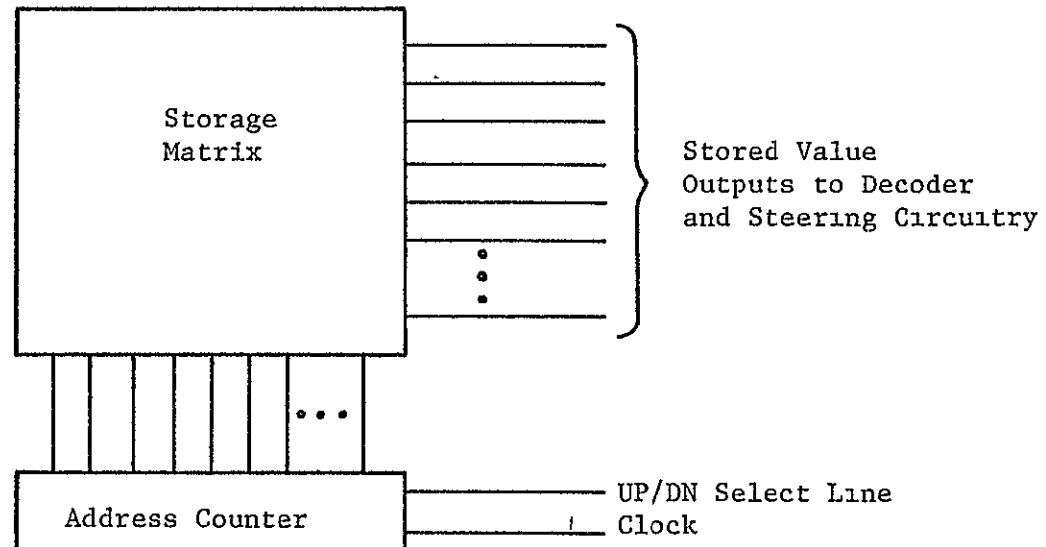


Figure 5-22. Illustration of Counter Arrangement to Implement a Mirror Image Coefficient Matrix

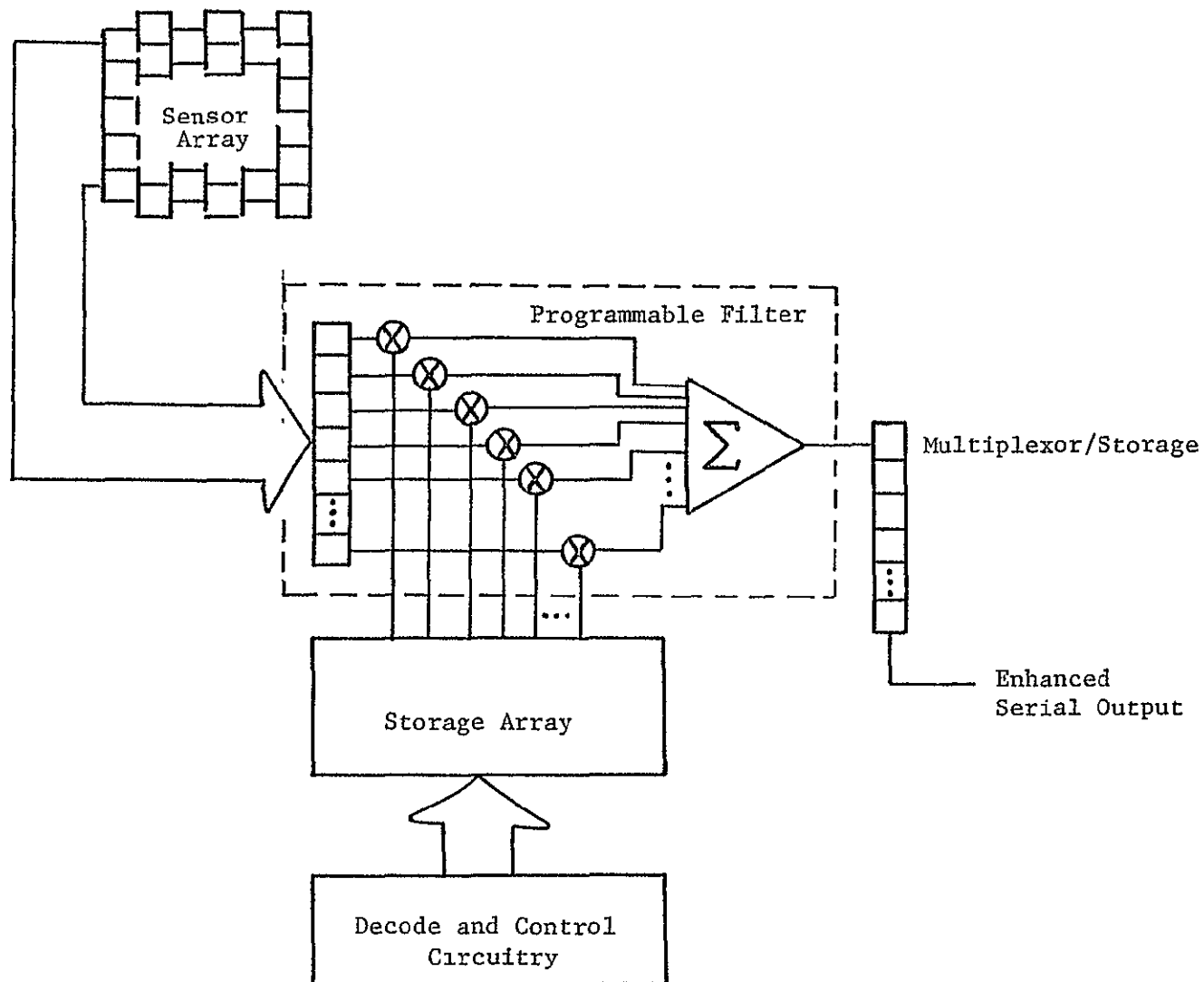


Figure 5-23. Block Diagram Illustration of Microcoded Processing Array Utilizing a Programmable Filter and Storage Array

U.2

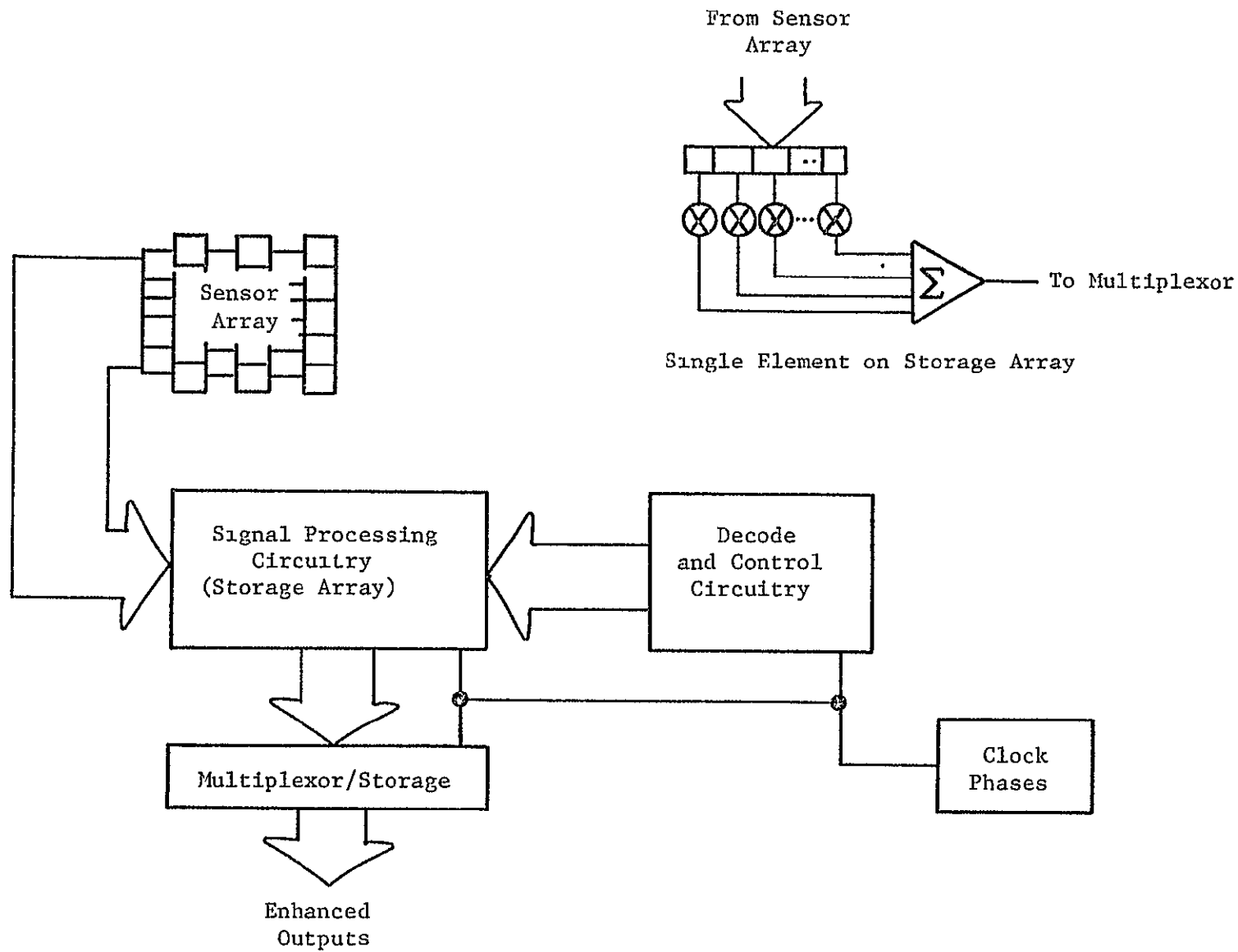


Figure 5-24. Block Diagram Illustration of Microcoded Processing Array and System with Single Storage Element Detail

tap weights shift cells, including the MOSFET structure, shall have an area kA , where k is some constant multiple of the area A with a value greater than one ($k > 1$) due to the added MOSFET structure. Each current sensing differential amplifier shall have a constant area to be designated by the value D . Area between shift cells in a parallel loading scheme shall be assumed to be negligible.

The design examples shall be presented using total area as a criteria, thus assuming that the trade-offs between circuit complexity and processing speed are apparent. The sensor area and input/output circuitry shall be ignored because these may be chosen independent of the processing circuitry.

Retracing the suggested designs and beginning with a parallel structure involving a single fixed tap weight transversal filter per row it can be seen that each shift cell in each filter corresponds to an element in the pseudoinverse column. The area of the signal processing circuitry as depicted by Fig. 5-15, can be expressed by

$$\text{TOTAL AREA } A_{t1} \approx (rc)A + rD \quad (5-24)$$

where r and c are the number of rows and columns, respectively, of the pseudoinverse M^+ matrix. The next design employed a programmable transversal filter with (rc) absolute storage values, assumed to be analog fixed tap weight storage elements. Assuming that the decoding and control circuitry is negligible in comparison to the storage area (less than 10%), the area (design as in Fig. 5-25) is apparently

$$\text{TOTAL AREA } A_{t2} \approx 2(rc)A + k(cA) \quad (5-25)$$

The 1st designs involve microcoding schemes of which there were several variations. The first scheme illustrates the microcoding of each major submatrix as a mirror image (Fig. 5-26). Decoding and control circuitry directs the loading of the programmable filter for each major submatrix.

$$\text{TOTAL AREA } A_{t3} \approx (Bn)A + H + 4kcA \quad (5-26)$$

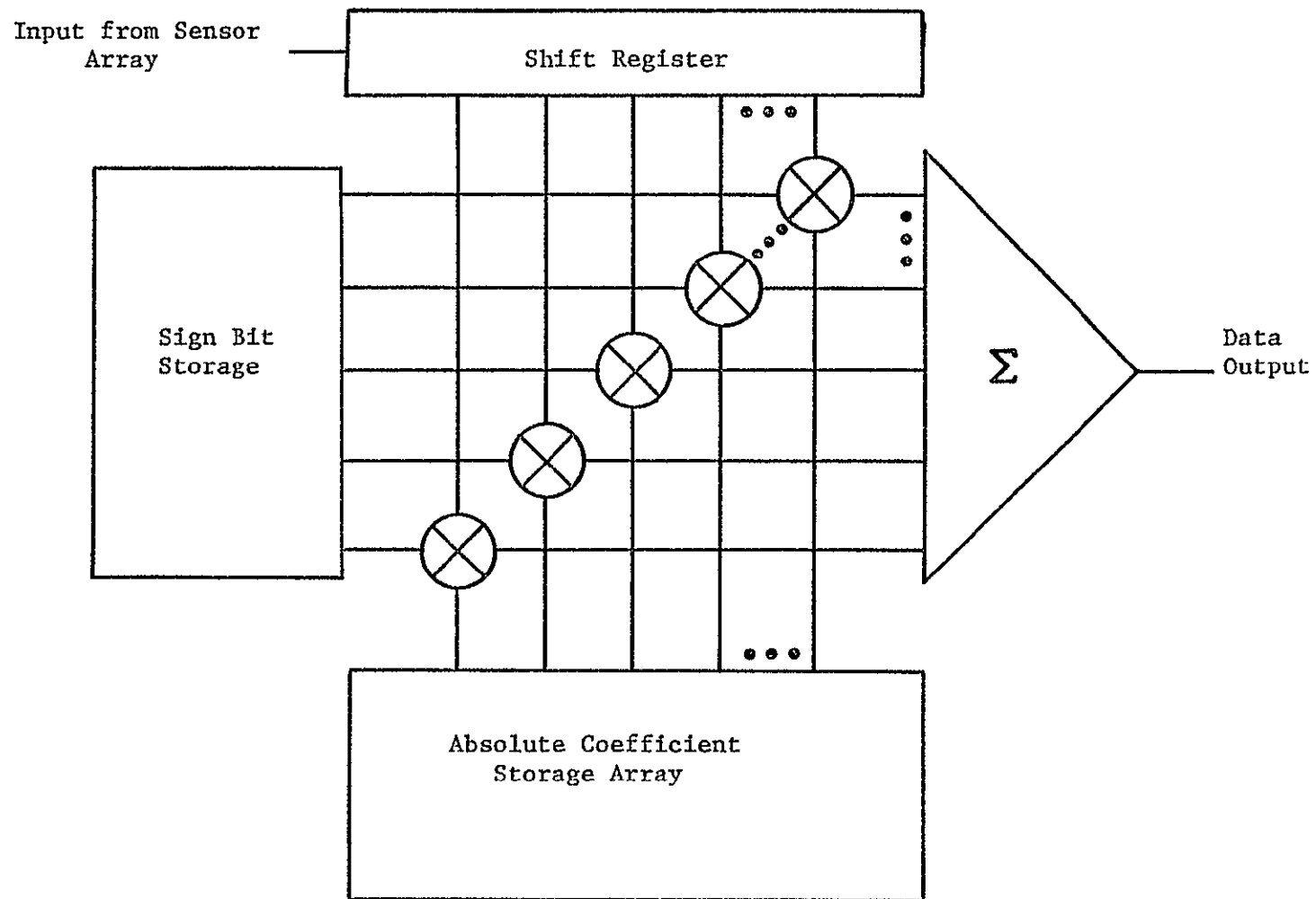


Figure 5-25. Illustration of Design for Equation A_{t2} .

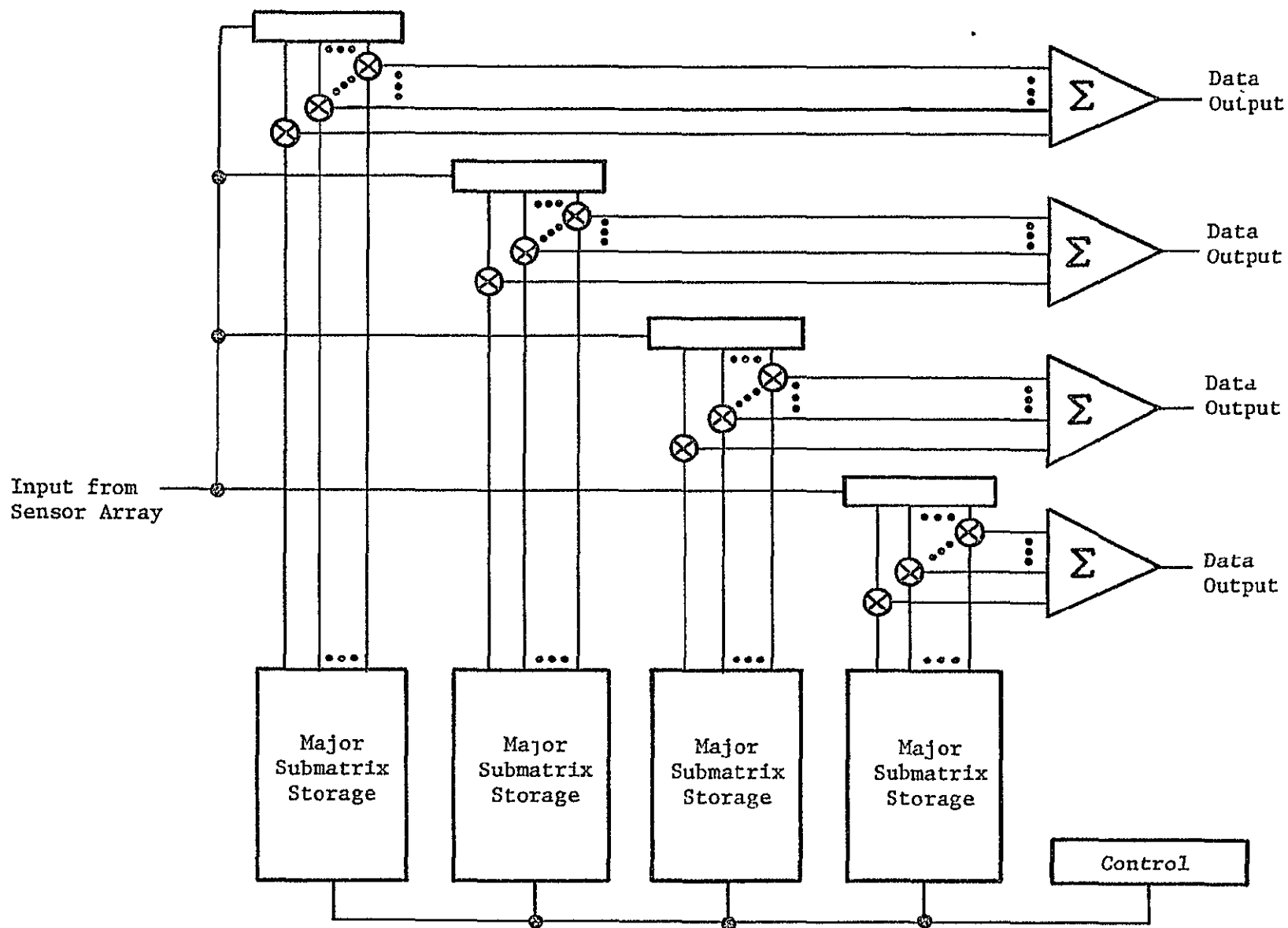


Figure 5-26. Illustration of Design for Equation A_{t3} .

represents such a scheme with a 4:1 enhancement where H is defined as the area due to the decoding and control circuitry, n is the number of sensors, and B the sum of the distinct row values per major submatrix over the complete sample. Attempting to characterize the best of the microcoding schemes, the last design discussed (refer to Fig. 5-24) shall be represented by

$$\text{TOTAL AREA } A_{t4} \approx [(Bn)A + BD] + H \quad (5-27)$$

where the difference between (5-26) and (5-27) is the deletion of the programmable filter. This is accomplished by the utilization of the storage elements as the processing circuitry.

The four design equations presented represent four different system implementations with their consequential trade-offs. Since various considerations are encountered in each design resulting in different variables, a cross comparison would be difficult to make. However, a tabulation of the design equations is presented in Table 1 as well as an estimated area for a given number of sensors capable of being processed in a 4:1 resolution enhancement. The area estimations are based on intuitive trends evidenced in examples of each design conducted with a small number of sensors and on an assumed constant cell area. It can be seen from Table 1 that the microcoding scheme (A_{t4}) demands the least area for the number of sensors.

If larger arrays are to be processed than can be easily accommodated in a single device, then another type of staging would be required. Attempting to realize the "smart sensor" concept, the combination of the microcoding technique with the staging approach depicted in Fig. 5-27 could make such a realization possible. The additional circuitry to process the overlapping external device sensors is easily justified by the trade-offs in information processing capability versus practical implementation size. Such a combination of techniques should remain useful regardless of future technology density improvements.

Error Considerations

Having seen several practical implementations of the resolution enhancement scheme in designs realized by charge-coupled devices, attention

is now focused upon the system's liabilities, particularly in the form of error. Reviewing the previous discussion on the error inherent in McRee's technique, the error shall begin with the error displayed in the system-operating algorithm.

From (5-9), it can be seen that any initial error would propagate and accumulate through to the completed sample. The boundary conditions imposed by McRee to facilitate matrix inversion was removed by Henderson in his algorithm utilizing a least squares estimation technique to generate a pseudoinverse which contains the system design values. Another important limitation removed by Henderson [3] was the accountability for a noisy input which although assumed to approach zero for the realization has practical significance. However, if noise is present in the input, it could be characterized in the system input and will not be considered here. Henderson has experimentally determined the error for his algorithm to range from maximum of eight percent to a negligible minimum. However, the system on which the measurements were made incorporated analog-to-digital converters to quantize the sensor outputs before being fed into the processing system. Henderson has noted that by increasing the number of quantization levels the accuracy improved significantly. The implication of this apparent trend is that quantization error is a significant contributing factor to the overall system inaccuracy. If the estimation error is considered to be insignificant, then quantization error is the only remaining error source in the system algorithm. As mentioned before, if CCDs are utilized as a transversal filter, then analog sensor output is processed thereby reducing system error to the practical limitations of the device. Elimination of the conversion units (a/D) has two definite benefits, namely, to reduce system complexity and to significantly improve system accuracy. With this in mind, prominent CCD limitations and error sources should be considered, and, since discussions of error in CCDs have been published (17, 18, 19, and 20), only a brief survey will be included here.

Commencing with the most significant source of error, transfer inefficiency ϵ , it can be seen that, as a packet of charge is transferred through hundreds of cells, that some charge carriers would remain trapped

in substrate imperfections or be lost to recombination centers. As can be expected, the transfer inefficiency ϵ would be a function of clock frequency and also of device geometry; therefore, each device designed would have a different value ϵ with current ϵ approaching 10^{-5} . Another source of error would be that generated by nonlinearities in the charge injection and charge sensing, with the last prominent error source to be that of thermal generation. Injection and sensing nonlinearities can be treated in various ways as a trade-off between complexity and size of the circuitry versus nonlinearity. Thermal generation is also termed dark current and is most prominent when low clock rates, high temperatures, or numerous shift cells are encountered. All the prominent error sources can be compensated for by several design techniques with the accompanying trade-offs. There are other error sources beyond the device level, namely those encountered in device fabrication, such as photolithography resolution, all of which should be reduced with evolving technology.

A more definite characterization of error should be undertaken after an experimental device has been fabricated.

Conclusions and Recommendations

Several designs are presented using CCD technology to implement an unique enhanced resolution algorithm. System errors are expected to be largely dependent upon device error inherent in the fabrication except for sensor and algorithm error. It has been shown theoretically that CCD technology can produce a significant reduction in processing circuitry complexity (e.g. elimination of A/D conversion). A further savings could be realized by use of microcoding techniques as shown by Table 1. The last design presented (A_{L4}) indicates the best circuit economy even with conservative assumptions. Processing speed is a design trade-off between resolution enhancement, target or sensor maximum velocity, and circuitry complexity. However, processing time is expected to be in the tens of microseconds for a single sample conversion. Experimental design and fabrication of devices should be undertaken to help define the best number of sensors for a building block or staged approach. Complex trade-offs between number of leads, redundant circuitry percentage, and other practical considerations make experimental work desirable.

TABLE 1

Designated Design	Design Equations	Estimated Parameters for Five Sensors (square units)	Estimated Parameters for Three Sensors (square units)
A_{t1}	$(rc)A + rD$	$900A + 36D = 1044$	$144A + 16D = 208$
A_{t2}	$2(rc)A + (kc)A$	$1800A + 25kA = 1850$	$288A + 9kA = 306$
A_{t3}	$(rc)A + D$	$900A + D = 904$	$144A + D = 148$
A_{t4}	$(Bn)A + BD + H$	$70A + 14D + H = 378$	$24A + 6D + H = 96$

- Assumptions:
- 1) $D = 4A$
 - 2) $H = (Bn)A + BD$
 - 3) $k = 2$
 - 4) $A = 1$

Table 1. Compilation of Design Equations and Estimated Results for Three and Five Sensors with Assumptions of Support Circuitry Size.

The sensor array is assumed to sense and retain a sample of a displayed energy density field (e.g. light). By varying a threshold such that the value above the threshold is processed, a vertical binary mosaic representing the energy density can be formed. Preliminary studies showed that in this manner gray-level information could be retained, but to an unknown degree of resolution. A vertical binary mosaic using a previous example is displayed in Fig. 5-28. However, it can be seen that a large degree of error is involved which is suspected to be due primarily to quantization.

With the ability to preserve gray-level information and the inherent angle information due to the sampling scheme and motion, another form of information may be available. By properly correlating the information input into the system, a pseudo-dimensionality may be achievable giving rise to "depth" information. Such speculations are reserved though future work.

In either case, it can be seen that the processing of analog information is highly advantageous, particularly from the low error viewpoint.

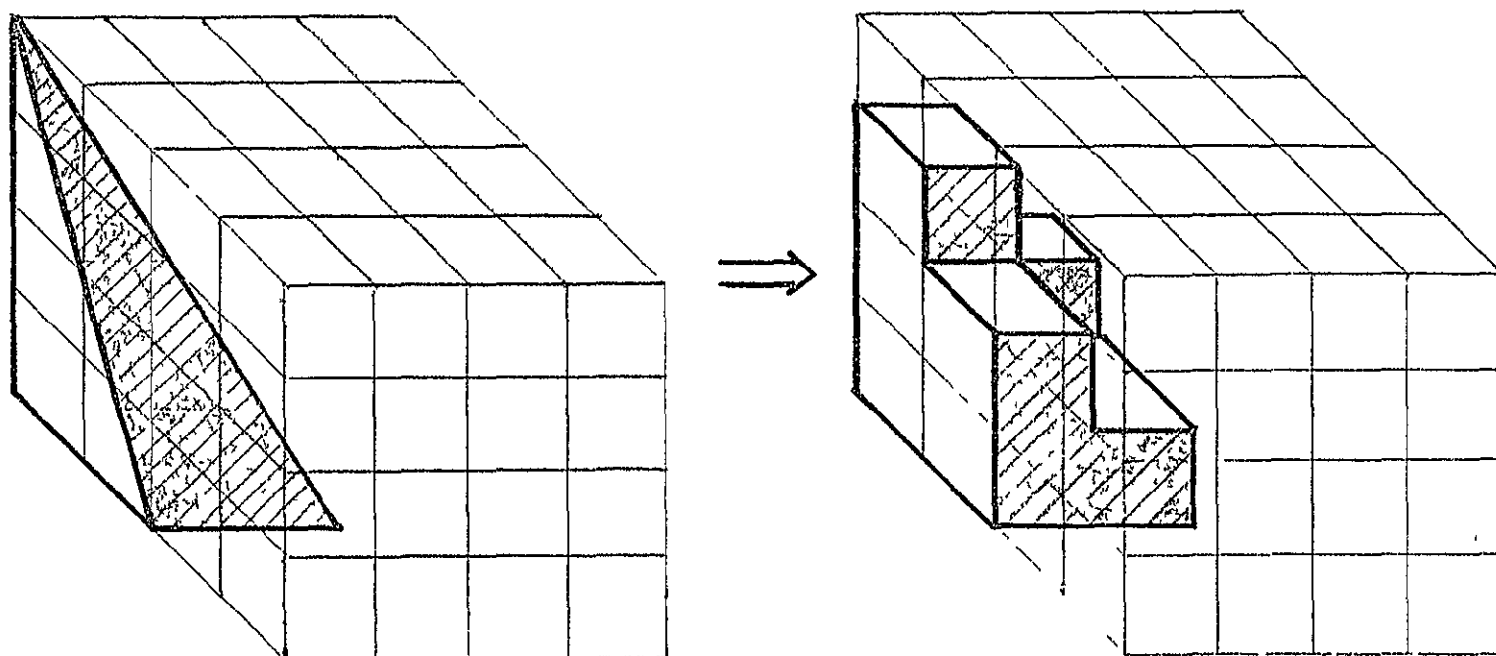


Figure 5-28. Example Transformation of Various Thresholds Resulting in a Pseudo-dimensionality.

REFERENCES

1. T. A. Irving, Jr., Sajjan G. Shiva, and H. T. Nagle, Jr., "Flip-Flops for Multiple-Valued Logic," IEEE Trans. on Comp., Vol. C-25, No. 3, Mar. 1976.
2. G. J. McRee and E. S. McVey, "A Method for Increasing the Resolution of Discrete Output Optical Sensors," Proc. of the 9th Annu. IEEE Reg. III Conf., Apr. 1971.
3. K. C. Henderson and E. S. McVey, "A Method for Increasing the Resolution of Discrete Optical Sensors," IEEE Trans. Ind. Elct. & Cont. Instru., Vol. IECI-21, No. 2, May 1974.
4. Carlo H. Sequin and Michael F. Tompsett, Charge Transfer Devices, Academic Press, Inc., 1975.
5. R. W. Brodersen, C. R. Hewes, and D. D. Buss, "A 500-Stage CCD Transversal Filter for Spectral Analysis," IEEE Trans. on Electron Dv., Vol. ED-23, No. 2, Feb. 1976.
6. John Mattern and D. R. Lampe, "A Reprogrammable Filter Bank Using Charge-Coupled Device Discrete Analog-Signal Processing," IEEE Trans. S-S Cir., Vol. SC-11, No. 1, Feb. 1976.
7. M. H. White and W. R. Webb, "Study of the Use of Charge-Coupled Devices in Analog Signal Processing," Final Rprt., Westinghouse Def. & Elct. Sys. Ctr., N00014-74-C-0069 Item A002, May 1974.
8. D. D. Buss, D. R. Collins, W. H. Bailey, and C. R. Reeves, "Transversal Filtering Using Charge-Transfer Devices," IEEE Jnl. S-S Cir., Vol. SC-8, No. 2, Apr. 1973.
9. D. D. Buss, W. H. Bailey, and A. F. Tasch, "Signal Processing Applications of Charge Coupled Devices," CCD74 Int. Conf. Edinburgh, Proc., 1974.
10. A. Ibrahim, L. Sellars, T. Foxall, and W. Steenart, "CCDs for Transversal Filter Applications," Tech. Dig., Int. Electron Dv. Mtg., Washington, D.C., Dec. 1974.
11. A. L. Lancaster and J. M. Hartman, "A recirculating CCD with Novel Input and Output Structures," Tech. Dig., Int. Electron Dv. Mtg., Washington, D.C., 1974.
12. R. W. Brodersen, D. D. Buss, and A. F. Tasch, Jr., "Experimental Characterization of Transfer Efficiency in Charge-Coupled Devices," IEEE Trans. Electron Dv., Vol. ED-22, No. 2, Feb. 1975.
13. Hamdi El-Sassi and R. S. C. Cobbold, "On the Decay of Small Amounts of Signal Charge in Surface-Channel CCDs," IEEE Trans. Electron Dv., Mar. 1975.

14. G. A. Clough, W. Kotyczka, and D. M. Erb, "Experimental Observation of Signal Degradation in Recirculating Charge-Coupled Devices," Tech. Dig., Int. Electron Dev. Mtg., Washington, D.C., 1974.
15. C. H. Sequin and A. M. Mohsen, "Linearity of Electrical Charge Injection in Charge-Coupled Devices," IEEE Jnl. S-S Cir., Vol. SC-10, No. 2, Apr. 1975.

APPENDIX

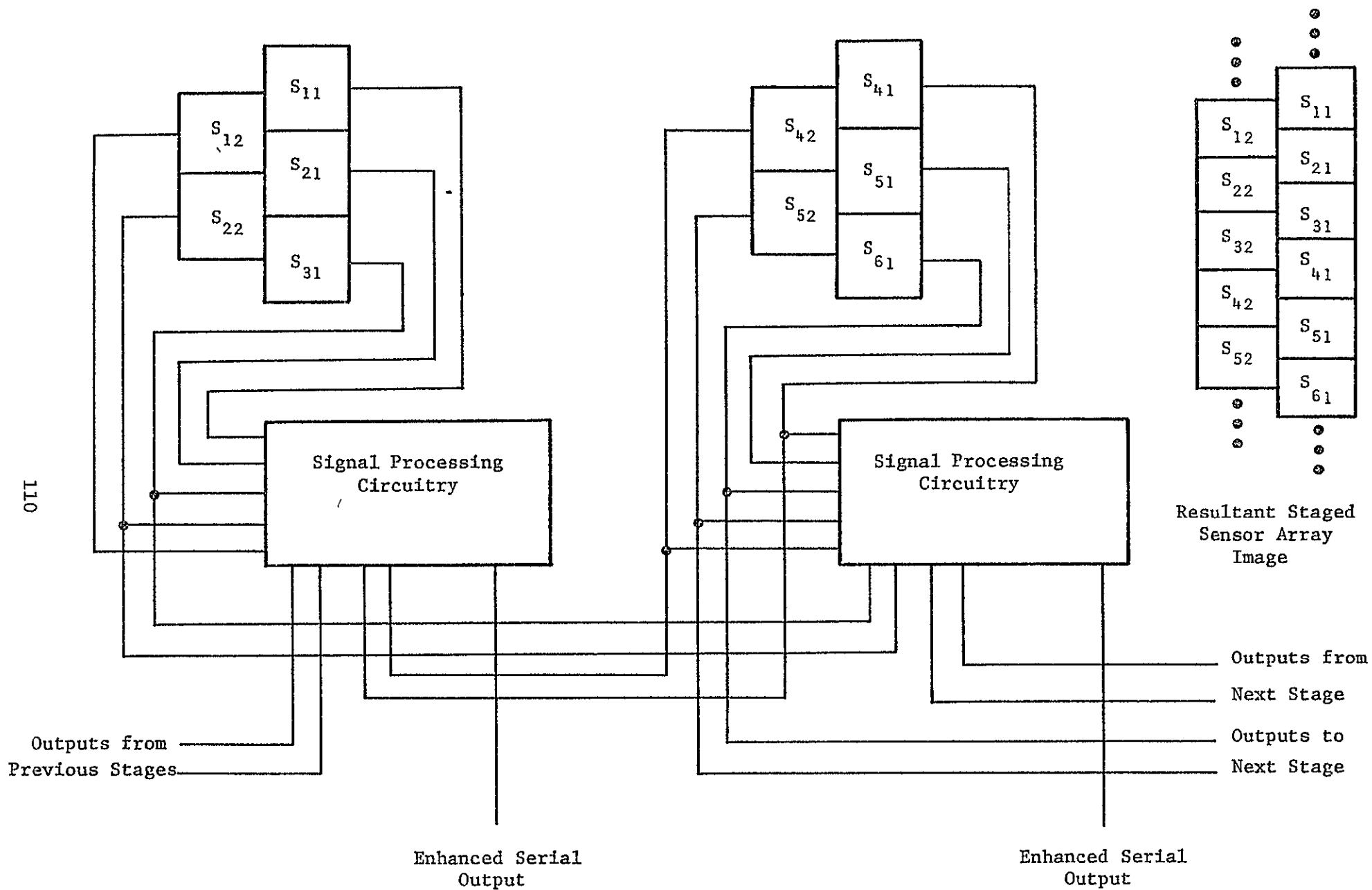


Figure 5-27. Illustration of Staged Staggered Sensor Column Array to Show Formation of Larger Arrays.

CHARGE-COUPLED DEVICES APPLIED TO CONTROL AND PATTERN RECOGNITION SYSTEMS

E. S. McVey

E. A. Parrish *

Abstract

The application of CCD (Charge-Coupled Device) processors to pattern recognition and control systems is suggested in this paper. Specific implementations for control systems are presented for linear time invariant, linear time varying, and nonlinear systems. Example structures are also presented for pattern recognition applications. Estimates of processing times are provided

* Both with the School of Engineering and Applied Science, University of Virginia. The work on which this paper is based was supported by the National Aeronautics and Space Administration, Langley Research Center, under Grant

Number NSG 1223

*PRECEDING PAGES NOT Pages 111 - 117
REQUIRED - F 2/18/77*

ORIGINAL PAGE IS
OF POOR QUALITY

PRECEDING PAGE BLANK NOT FILMED
PRECEDING PAGE BLANK NOT FILMED

Introduction

Recently, a new form of charge transfer device called the CCD (Charge-Coupled Device) has been developed which processes discrete analog samples of data. The inherent manufacturing simplicity, small size, and low power allow the use of many stages in a single unit. This, coupled with their fast speed, makes practical the processing of extremely larger quantities of data in applications such as image processing. Although devices are not yet commercially available, families are expected to emerge soon, thus, it is timely to consider their application in engineering systems

The initial discovery of CCD's was announced by Bell Telephone Laboratories on May 9, 1970 [1]. Since then, commercial devices for optical signal detection, such as discrete element sensors [2], and for digital memories [3], have been marketed.

A charge-coupled device is a semiconductor component [4] capable of storing quantities of charge (called packets) and transferring them between adjacent cells. This, in effect, becomes a discrete analog delay line or shift register. The addition of nondestructive readout taps, along with tap weights and on-chip summing, provides an analog processing capability of unique potential.

Another charge transfer device, which was the forerunner to CCD's, is the Bucket Brigade Device (BBD) [5]. The BBD consists of a series of field effect transistors in which charge packets are transferred from one cell to another. The BBD's are of less value for sophisticated signal processing and, thus, will not be considered in the sequel.

For a detailed explanation of CCD characteristics and operation the reader is referred to [6]. Essentially, they are composed of a substrate of

semiconductor material on which a thin insulation layer is placed. The cells in which the charge packets are stored are formed by depositing metal electrodes onto the insulator. This forms capacitors which operate analogously to the capacitors and switches shown in Figure 1. When any switch closes, charge is transferred to an adjacent capacitor, and then the switch is opened. With proper synchronization between switches, charge transfer can be controlled. Of course, this representation is crude; but it is a convenient approximation of operation.

In physical electronic terms, when the electrodes are properly biased with respect to the substrate, potential wells are formed in which the charge is stored. The charge transfer action is produced by changing the potentials associated with adjacent cells. Charge is injected into the input cell through a conventional junction diode. Operation of this diode in conjunction with the first cell allows sampling of the analog input signal, yielding a discrete analog representation.

As might be expected, charge cannot be transferred between cells without loss, since a fraction of the charge will be left behind, thus causing a reduction in the signal-to-noise ratio. Typically, the transfer inefficiency ϵ is on the order of 10^{-4} to 10^{-5} at 1 MHz and increases rather rapidly at higher frequencies.

The unique discrete analog signal processing capabilities of CCDs may make practical pattern recognition systems which, until now, have been only theoretical possibilities because of the extremely large quantities of data to be processed. Also, control system applications appear to be practical in systems of high order, such as where kalman filtering is required.

ORIGINAL PAGE IS
OF POOR QUALITY

Basic Structures

The most basic structure is a serial in/serial out (SI/SO) discrete analog delay line, which is shown in Figure 2a. The data are transferred from left to right at a rate determined by the clock, with a period T . The maximum time for a sample to be in residence (delayed) is NT . Delays of 0.1 to 3 seconds at room temperature are reported in the literature and calculations indicate that much longer times are possible if the temperature is reduced. There are many applications for this basic structure. It is also used as a building block in more complex components.

A serial in/parallel out (SI/PO) structure is shown in Figure 2b. The input signal is sampled and shifted as in a SI/SO device, but the content of each cell is available on a nondestructive readout basis. If tap weights are used in conjunction with current summing, it is possible to implement a number of useful operations which are considered later.

The parallel in/serial out (PI/SO) configuration is shown in Figure 2c. It can be used for applications such as multiplexing.

The structure of Figure 3 is likely to become a basic building block in the near future. As shown, it consists of a SI/SO structure with tap weights and a summer as mentioned above, all on a single chip. The current summing is easily and accurately accomplished because of the basic way the structure operates. It operates as a nonrecursive or transversal filter, whose output at the n^{th} clock period is the weighted sum of the previous k samples,

$$g(n) = \sum_{i=0}^k h_i x_{n-i} \quad n \geq k \quad (1)$$

For example, if the tap weights h_i in (1) define samples of the time inverse of a desired signal, then the CCD processor of Figure 3 is a matched filter. Or,

if the tap weights are set equal to samples of a given weighting function, then the structure can perform a discrete convolution. Finally, vector inner products can be computed by reading the output only at $n = k$, so that

$$g(k) = \sum_{i=0}^k h_i x_{k-i} \quad (2)$$

By changing the order of the weights in Figure 3, the more familiar form

$$g = \sum_{j=0}^k h_j x_j \quad (4)$$

is obtained, which can be recognized as a linear discriminant function [7], a very useful tool in pattern recognition.

Fixed tap weight transversal filters are also being used to implement frequency domain processing. For example, by employing the Chirp z transform [8], CCDs can be used to implement discrete Fourier transforms which compute a 500-point power density spectrum [9]. By using CCD Fourier transform processors, convolution of two arbitrary waveforms can be accomplished through frequency domain multiplication.

Another form of CCD processor is under investigation which could prove to be very valuable to the structure of future instrumentation systems. As shown in Figure 4, this filter allows the tap weights to be programmed electrically, thus making it applicable to adaptive control, trainable pattern classifiers, etc. The variable tap weight transversal filter is difficult to implement and will probably not be available for some time. Nevertheless, it could be very important to the development of new system architectures and so deserves attention even now.

ORIGINAL PAGE IS
OF POOR QUALITY

Control System Applications

Control systems can use CCD devices to make modeling calculations or for realizing required transfer functions [10]. State space models are an example where matrix-type calculations are required. Suppose, for example, a linear time invariant system containing a Kalman filter is to be implemented. Then, calculation of states at time $k + 1$ based on states at time k must be made using the state transition matrix. The unforced system differential equation is

$$\dot{x} = Ax \quad (5)$$

for which the solution is

$$x(t) = e^{A(t-t_0)} x(t_0) \quad (6)$$

For the discrete case of interest here, (6) becomes

$$x[(k + 1)T] = \phi(T)x[kT] \quad (7)$$

where x is an M -dimensional vector, A is an $n \times n$ matrix, T is the sampling period, ϕ is the $n \times n$ state transition matrix, and k is a positive integer.

As is well known, (7) takes the general form

$$x[(k + 1)T] = \begin{bmatrix} \phi_{11} & \phi_{12} & \cdot & \cdot & \cdot & \phi_{1n} \\ \phi_{21} & \phi_{22} & \cdot & \cdot & \cdot & \phi_{2n} \\ \cdot & & & & & \\ \cdot & & & & & \\ \phi_{n1} & \phi_{n2} & \cdot & \cdot & \cdot & \phi_{nn} \end{bmatrix} x[kT] \quad (8)$$

where the elements of ϕ are constants for the time invariant cases being considered.

ORIGINAL PAGE IS
OF POOR QUALITY

Implementation of equation (8) is shown in Figure 5. Each transversal filter is equivalent to the basic structure of Figure 3, where the values of the tap weights are equal to the row components of the Φ matrix, i.e., for the top filter, h_k equals ϕ_{11} , h_{k-1} equals ϕ_{12} , etc. This assures that the first component of the state, $x_1(kT)$, is fed in first, $x_2(kT)$ is fed in next, etc. A typical cycle time might be $n + 2$ clock periods to obtain the state $x(k + 1)T$ in storage in the PI/SO register from the initial value $x(kT)$.

The reader will recognize that this implementation can be used to simulate the homogeneous solution for any dynamic system. A complete simulation would include another matrix driven by the input plus appropriate summation, etc.

An alternate approach using transversal filters for control and simulation makes use of difference equation formulations. The general n -th order linear time-invariant difference equation is given by

$$y(k) = b_0 u(k) + b_1 u(k - 1) + \dots + b_p u(k - p) - a_1 y(k - 1) - a_2 y(k - 2) - \dots - a_n y(k - n), \quad k = 1, 2, \dots \quad (9)$$

where $y(k)$ is the output at time $t = kT$ and $y(k - 1)$, $y(k - 2)$. . . are the output values at $(k - 1)T$, $(k - 2)T$, etc. Similarly, $u(k)$ is the input at $t = kT$ and $u(k - 1)$, $u(k - 2)$. . . are the input values at times $(k - 1)T$, $(k - 2)T$, . . . , and the a_1 and b_1 are the system coefficients. This is a recursive equation involving both the input and past values of the output. Because of the need for additional on-chip amplifiers, CCD recursive filters may not be as cost effective as transversal filters. Thus, the suggested implementation of (9) makes use of external feedback paths around nonrecursive filters, as shown in Figure 6.

CCDs appear to offer much promise for real time simulation of large linear time invariant systems, because they can process large quantities of discrete analog data very rapidly. For time varying and nonlinear cases variable tap weight devices would seem to offer promise. Structures required for their implementation would be similar to those already presented, except that the filter in Figure 4 would be used.

Pattern Recognition Applications

The pattern recognition system to be considered is shown in Figure 7. The input waveform, $s(t)$, to be identified as belonging to one of C different classes, ω_k , $k = 1, 2, \dots, C$, is sampled to form a vector z as

$$z = (z_1, z_2, \dots, z_n)^T \quad (10)$$

and fed to the Feature Extractor.

The purpose of the Feature Extractor is to measure significant properties of the vector z so that the Classifier can make accurate decisions. This implies a data reduction transformation which removes much of the redundant information while compressing relevant detail into as few feature measurements as possible. Only linear transformations are being considered, so the Feature Extractor can be modeled by

$$x = Az \quad (11)$$

where z is $n \times 1$, the transformation matrix A is $d \times n$ and the feature vector x is $d \times 1$, with $d < n$. For example, among the transformations that could be used are the discrete Fourier and Hadamard transforms. The feature vector serves as input to the Classifier.

Under the assumption that similarity among different feature vectors can

be represented by some distance criterion, the function of the Classifier is to partition the feature space such that feature vectors from the different pattern classes w_k lie in distinct hypervolumes and can thus be discriminated. The extent to which this may be done determines the error rate for the system.

Once again, the discriminant functions which make up the Classifier are constrained to be linear, such as

$$g_1(x) = w_1^t x + w_{10}, \quad 1 = 1, 2, \dots, C \quad (12)$$

where $w_1 = (w_{11}, w_{12}, \dots, w_{1d})^t$ is the weight vector and w_{10} is the constant associated with the 1th class. This still leaves a large variety of problems for which CCD processors will be applicable.

To illustrate CCD processor organizations which may be applicable in pattern recognition systems, consider the following hyperthetical problem [11].

1. Number of sample points, $n = 200$
2. Number features, $d = 20$
3. Number classes, $C = 60$

The feature extraction transformation to be used was given by (11). For convenience, let $w_{10} = 0$ so that the Classifier consists of the discriminant functions

$$g_1(x) = w_1^t x \quad 1 = 1, 2, \dots, 60 \quad (13)$$

Extension to the case $w_{10} \neq 0$ can be done easily.

One way in which to implement (11) and (13) involves a structure in which all discriminant functions are evaluated simultaneously, without the need for separate feature extraction. Such a realization is suggested by a

two-dimensional CCD matrix array [12]. First, substitute (11) into (13) to obtain

$$g = WAz = Qz \quad (14)$$

where $g = (g_1(x), g_2(x), \dots, g_{60}(x))^t$, $W = (w_1^t, \dots, w_{60}^t)$ and Q is a 60×200 matrix

Figure 8 shows the use of a SI/PO CCD shift register to obtain samples of the waveform to be classified. The Q matrix corresponds to the analog weighting conductance array, represented by dots in the figure. The PI/SO CCD shift register is used to shift out the discriminant values to a serial comparator, which determines the maximum discriminant function and thus makes the classification. If it is assumed that the CCD registers are operated at 1 MHz, then it is estimated that a decision can be made in about 600 μ s [11].

Another possible system structure is shown in Figure 9 which utilizes a series-parallel organization involving variable tap weight CCD correlators, read only memories (ROMs) and digital-to-analog converters (DACs). Again, the Q matrix of (14) is used but now is stored in the ten Q matrix ROMs. Conceptually, each ROM contains six rows having 200 data each. First, the sample function to be classified is acquired and then one row at a time of the ten ROMs is loaded into the correlators. The operation performed is an inner product of a given row of the Q matrix with the z vector obtained from sampling the input, since no lags are involved. Ten discriminant values are obtained in parallel, requiring a total of six cycles of operation for the arbitrary number of DACs shown in the figure. Again, using a frequency of 1 MHz for the CCDs and inexpensive (slow) DACs, a decision will require about 13 ms [11].

Herein, of course, many variations of Figure 9, for example, in which a different number of DACs are used. In addition, it may prove advantageous in

some cases to provide separate structures for implementing the Feature
Extractor and Classifier, rather than combining them as in (14). The straight-
forward use of a number of fixed tap weight transversal filters, such as in
Figure 5, is also likely to be of value in some cases.

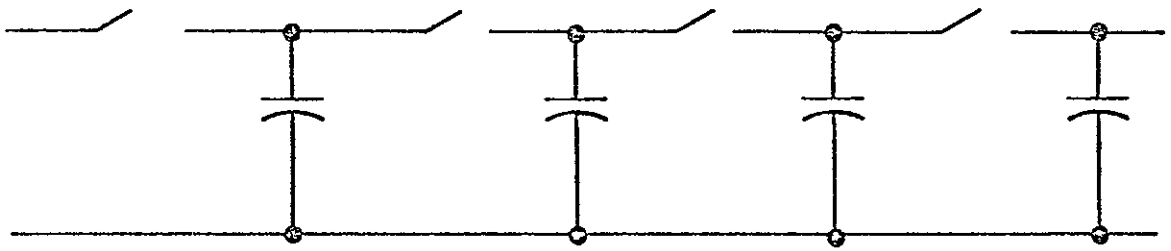
ORIGINAL PAGE IS
OF POOR QUALITY

BIBLIOGRAPHY

- 1 Electronics, p. 112, May 11, 1970
2. New Products, Electronics, April 17, 1975.
- 3 Altman, L., "Charge-Coupled Devices Move in on Memories and Analog Signal Processing," Electronics, pp. 91-101, August 8, 1974.
- 4 Kosonocky, W. F., "Charge-Coupled Devices - An Overview," 1974 WESCON Professional Program, Session 2, Los Angeles, California, September 1974
5. Butler, W. J., et al., "Practical Considerations for Analog Operation of Bucket Brigade," IEEE J Solid State Circuits, SC-6, pp. 391-395, December 1971.
- 6 Boyle, W. J. and Smith, G. E., "Charge-Coupled Devices - A New Approach to MIS Device Structures," IEEE Spectrum, pp 18-27, July 1971.
- 7 Nilsson, N. J., Learning Machines, McGraw-Hill: New York, 1965
- 8 Rabiner, L. R., et al., "The Chirp z Transform Algorithm," IEEE Trans on Audio and Electroacoustics, Vol. AU-17, No. 2, pp 86-92, June 1969
9. Biodersen, R. W., et al., "A 500-Point Fourier Transform Using Charge-Coupled Devices," Proc. of the Int'l. Solid State Circuits Conf, February 1975.
- 10 McVey, E. S. and Parrish, E. A., "The Application of Charge-Coupled Device Processors in Automatic Control Systems," submitted to IEEE Trans. on Automatic Control, 1976
- 11 Parrish, E. A. and McVey, E. S., "Implications of Charge-Coupled Devices for Pattern Recognition," accepted for publication, IEEE Trans on Computers, 1976
- 12 White, M. H. and Webb, E. R., "Study of the Use of Charge-Coupled Devices in Analog Signal Processing Systems," Final Report, Contract No W00014-74-C-0069, Westinghouse Defense and Electronic System Center, May 1975

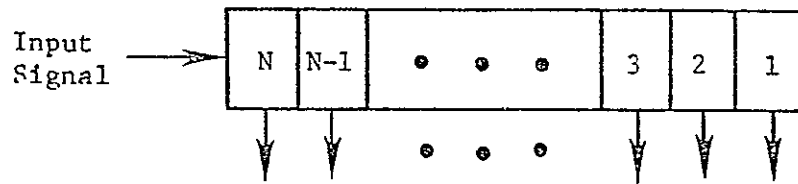
LIST OF FIGURES

- Figure 1 A Switch-Capacitor Analogy to CCD Operation
- Figure 2 Basic CCD Structures (a) SI/SO, (b) SI/PO, (c) PI/SO
- Figure 3 A Fixed Tap Weight Transversal Filter
- Figure 4 A Variable Tap Weight Transversal Filter
- Figure 5 A CCD Processor for Implementing (8)
- Figure 6 Implementation of a Linear, Time Invariant Difference Equation
- Figure 7 A Typical Pattern Recognition System
- Figure 8 A Possible Implementation of (14)
- Figure 9 A Series/Parallel CCD Structure for Implementing (14)

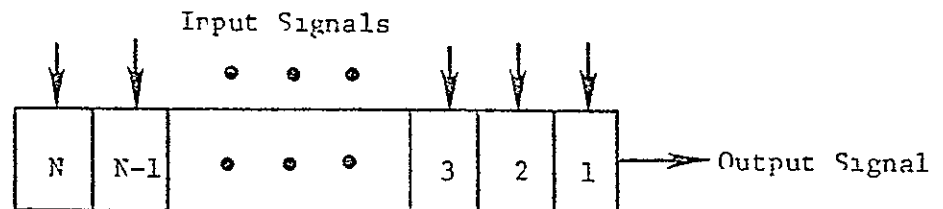




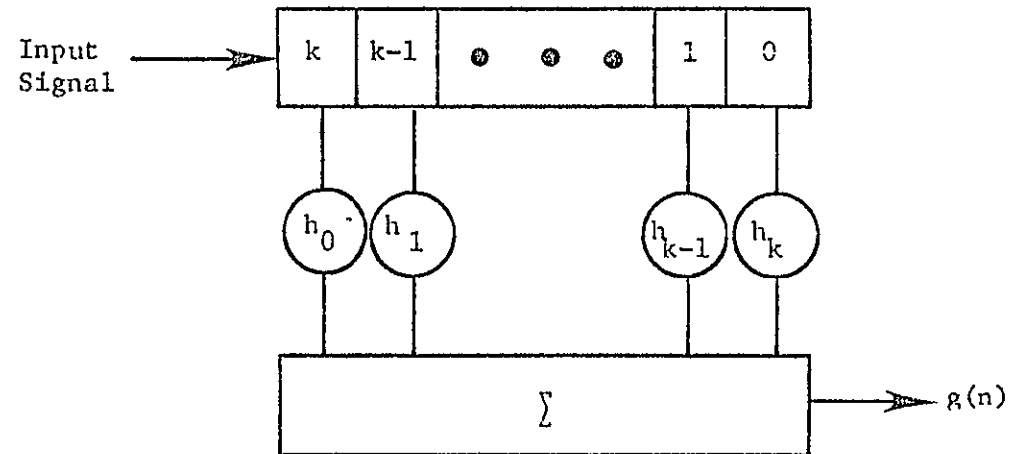
(a)

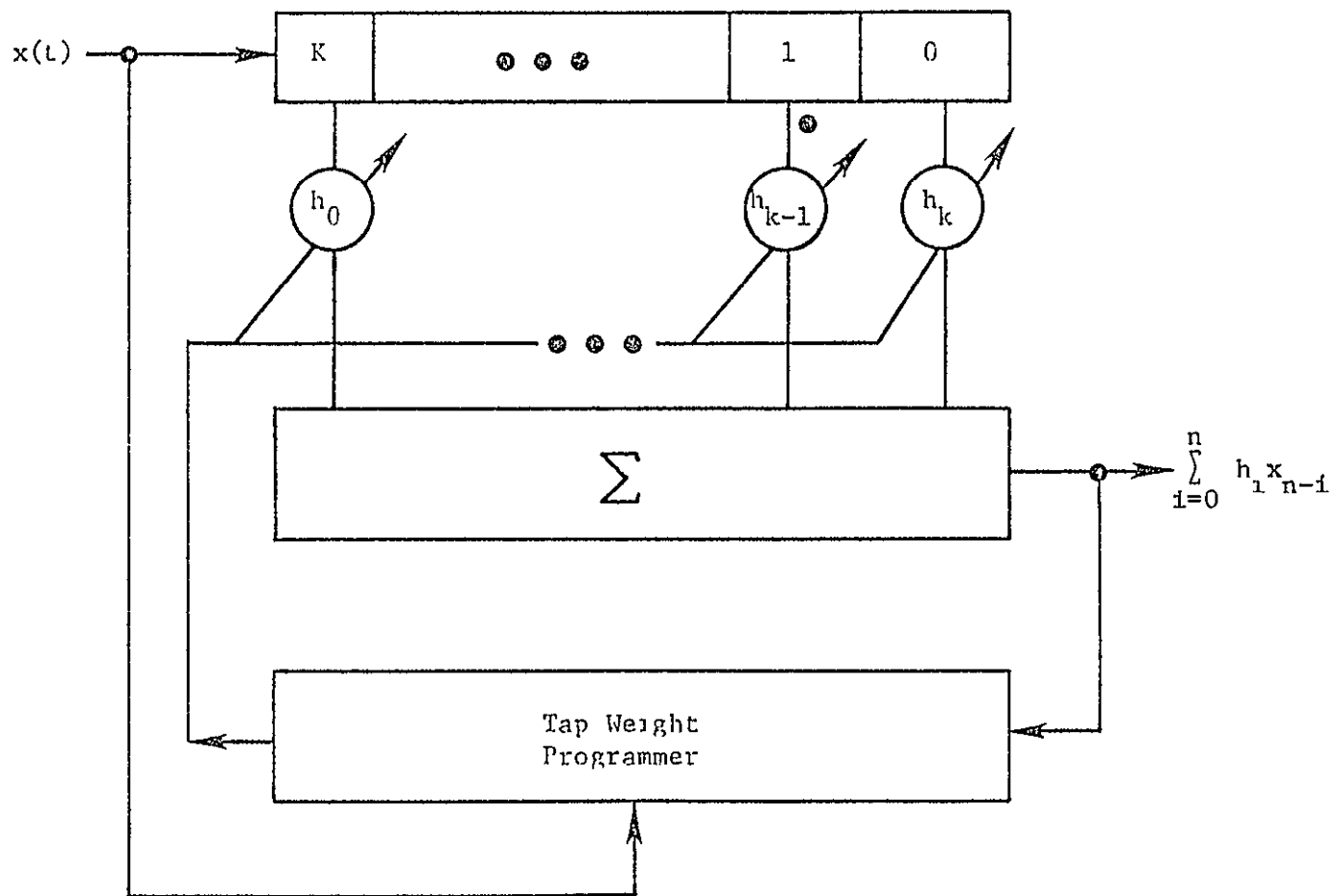


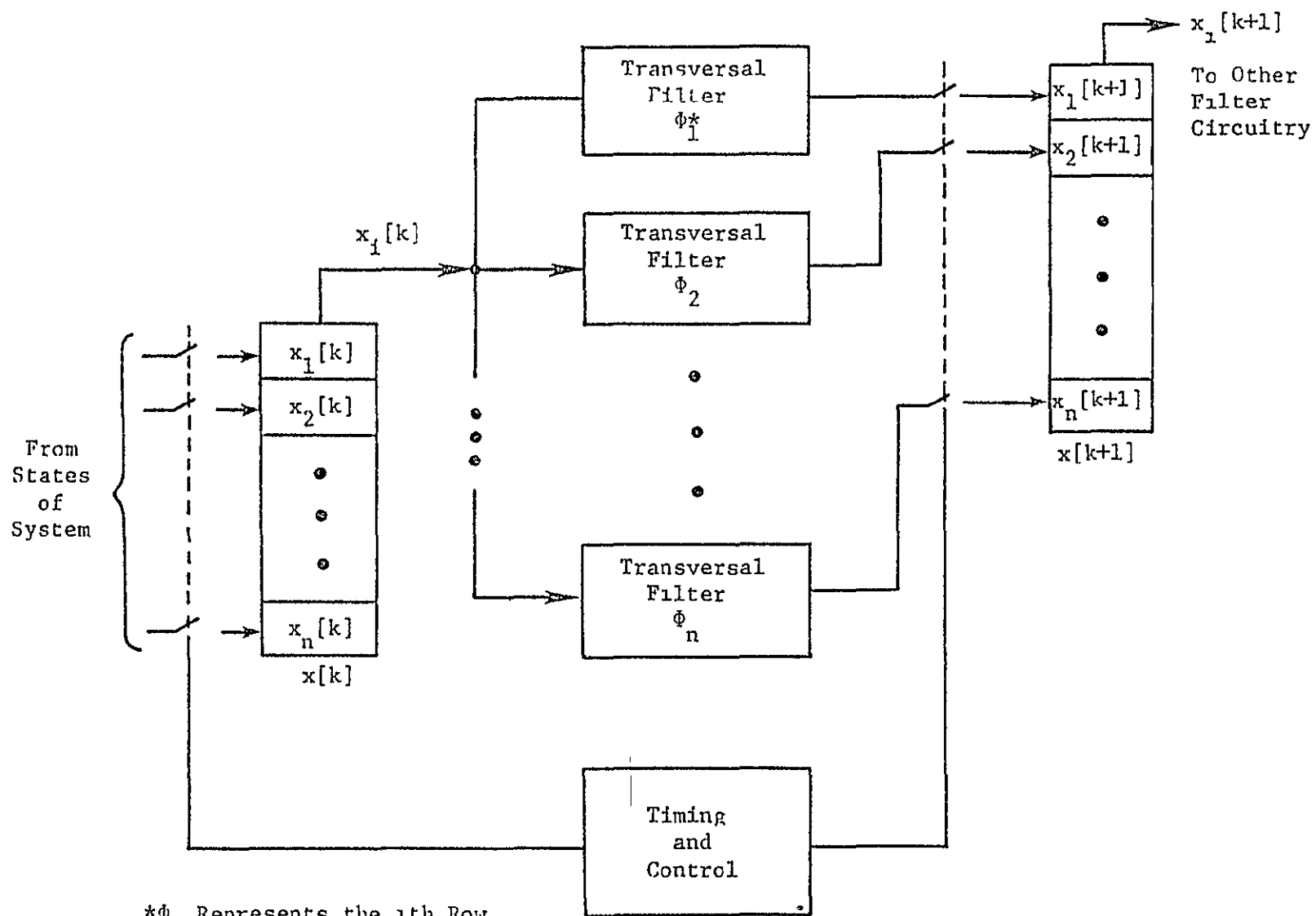
(b)



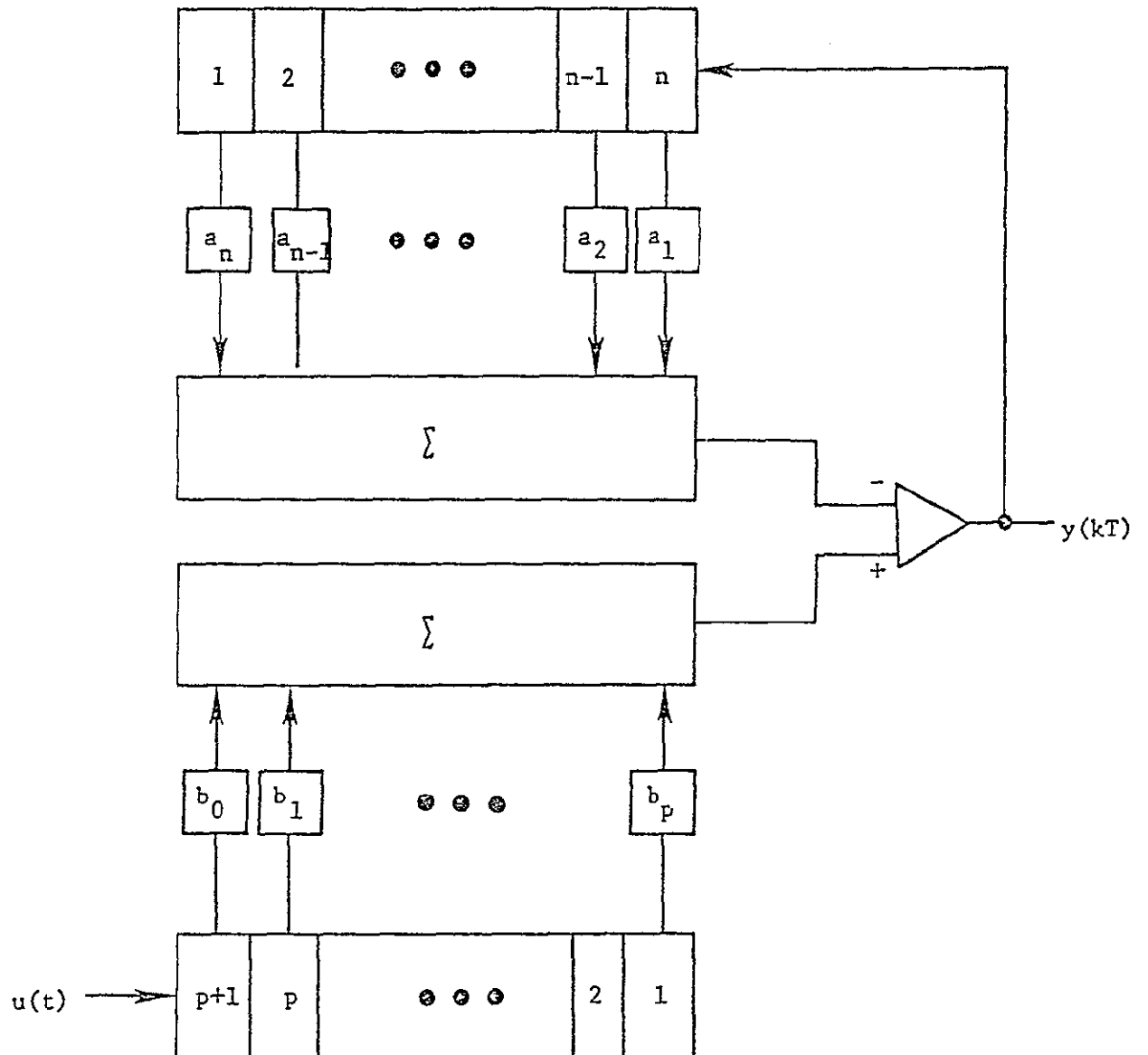
(c)

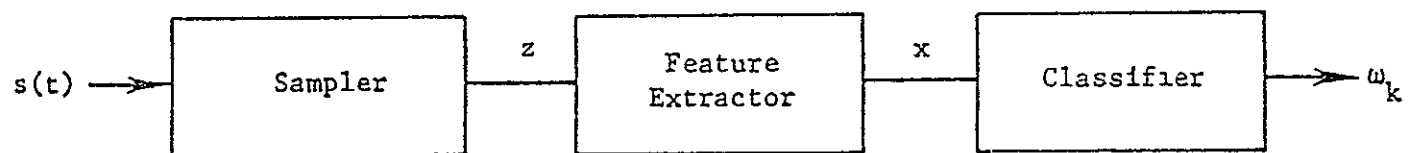


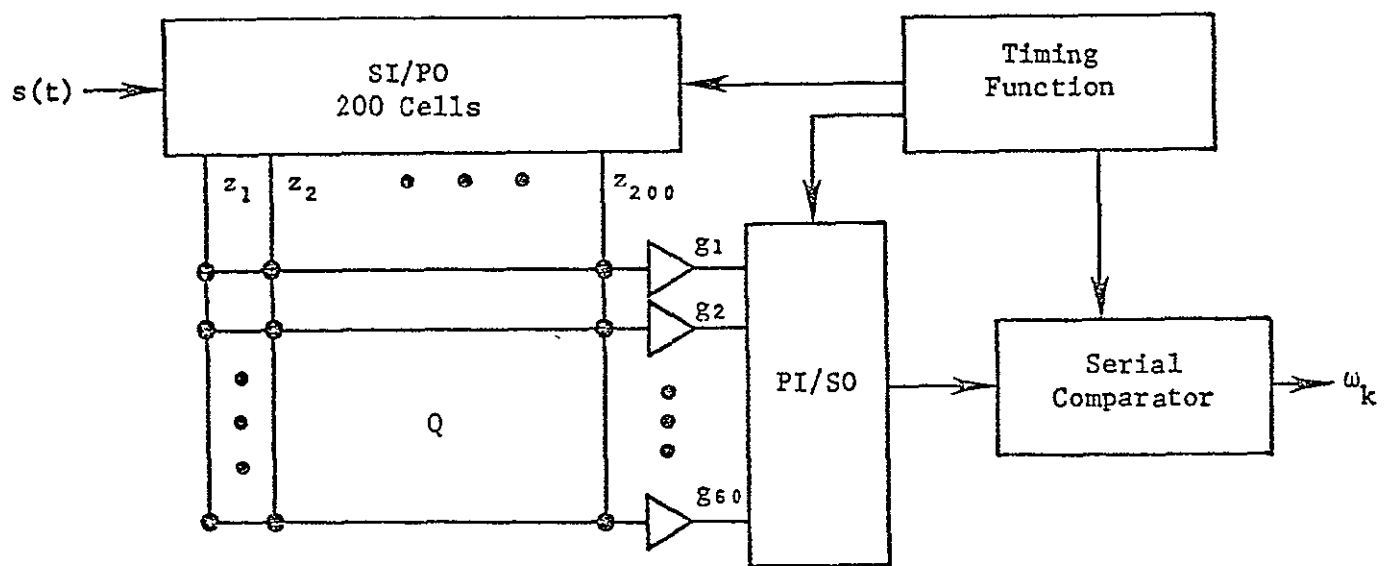


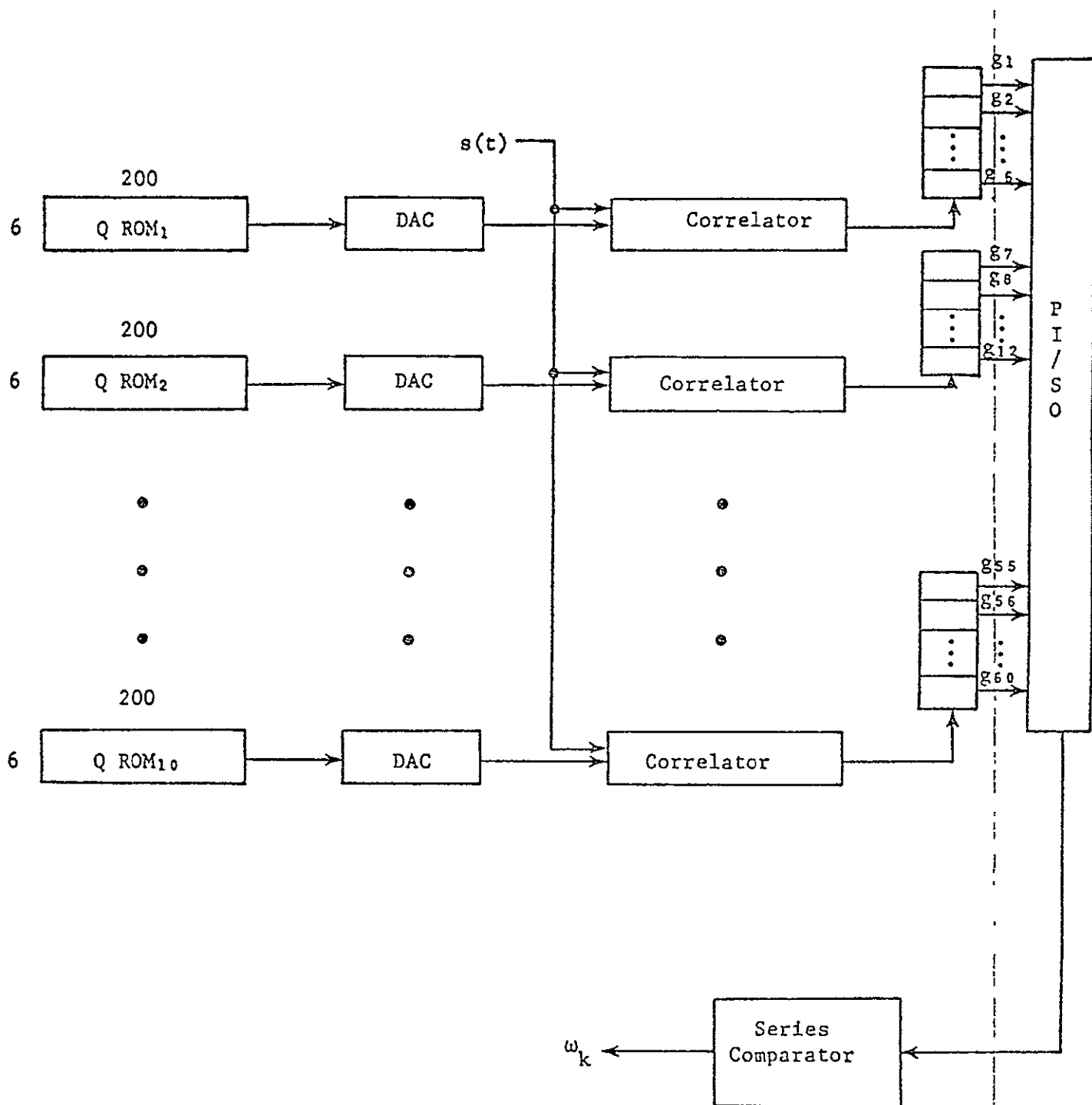


* ϕ_1 Represents the 1th Row
of the Transition Matrix Φ









THE APPLICATION
OF CHARGE-COUPLED DEVICE PROCESSORS
IN AUTOMATIC CONTROL SYSTEMS

E. S. McVey, Sr. Member, IEEE

E. A. Parrish, Jr., Member, IEEE

Abstract

The application of CCD (charge-coupled device) processors to automatic control systems is suggested. CCD processors are a new form of semiconductor component with the unique ability to process sampled signals on an analog basis. Specific implementations of controllers are suggested for linear time invariant, time varying and nonlinear systems. Typical processing time should be only a few microseconds. This form of technology may become competitive with microprocessors and minicomputers in addition to supplementing them.

ORIGINAL PAGE IS
OF POOR QUALITY

THE APPLICATION
OF CHARGE-COUPLED DEVICE PROCESSORS
IN AUTOMATIC CONTROL SYSTEMS

E. S. McVey, Sr Member, IEEE

E. A. Parrish, Jr., Member, IEEE *

Introduction

The development of CCD (charge-coupled device) processors offers control engineers new possibilities for implementing controllers. Charge-coupled devices are unique in that they make possible processing of discrete analog data. Their discovery was announced in May of 1970 by the Bell Telephone Laboratories [1, 2, 3], and a few commercial devices are available.[4] Although it is not yet practical to use CCDs in production equipment, except for applications involving memory and discrete image sensors, it is time to establish a foundation for their use in control systems. It is important that device researchers and manufacturers be made aware of the needs of control technology. It is the purpose here to suggest possible uses of CCD processors, delineate some of the expected problems and suggest possible areas of exploration.

A CCD is a semiconductor component[5] capable of storing quantities of charge called packets and moving them practically intact from one storage location to another. The packets represent the magnitude of signal samples. If the movement takes place at regular time intervals, a discrete delay line action is achieved.

The CCD is a member of the class called charge transfer devices of which the "bucket brigade device" (BBD)[6] is also a member. The BBD is older but of less potential value for control system signal processing than the CCD.[7] A number of papers are available which describe in detail the characteristics of CCDs, for example,[8]. They consist essentially of a substrate of semiconductor material, such as n-type on which is placed a thin layer of insulation. Cells are then formed by depositing metal electrodes on the insulating material. Biasing the electrodes with respect to the substrate creates a potential well in the semiconductor material. Charge injection is accomplished with a conventional PN junction. Charge may be transferred between adjacent cells by appropriately pulsing the associated electrodes. Although the transfer between cells is not perfect, it is adequate for many applications. The transfer inefficiency ϵ is on the order of 10^{-5} to 10^{-4} . [9] The ϵ depends on clock frequency, as one would expect. Present operation of these devices is usually restricted to less than 10 MHz. Transfer efficiency is important to analog data processing, because signal levels cannot be reestablished as in digital processing. The thermal generation of electron-hole pairs is a serious source of distortion, especially at elevated temperatures. This restricts storage in

Both with the School of Engineering and Applied Science, University of Virginia, Charlottesville, Virginia. The research on which this paper is based was supported by the NASA Langley Research Center on Grant No. NSG-1067 and Grant No. 1151.

a cell at room temperature to times less than one second. Storage time can be increased approximately by a factor of two for every 10°C decrease in temperature.

This technology has led to the development of discrete element sensors and TV cameras that are now available commercially.[10] It is anticipated that memories of very high densities will be developed. Small memories are now available commercially. Discrete analog CCD processors are not yet widely available, but families are expected to emerge soon.

Digital computers, especially small ones, are becoming widely used for control purposes. This seems to be a logical application also for CCD hardware. Different implementation possibilities are suggested in the following. It is emphasized that these ideas have not been proven experimentally, because it is not yet possible to obtain the devices. However, they are logical extrapolations based on successful experiments and on anticipated device developments.

Fundamental Configurations

Several CCD structures are under development which are considered basic building blocks. By using these devices with some additional circuitry, it is possible to implement linear transformations such as convolution, correlation and Fourier transforms.

The three basic building blocks are Serial In/Serial Out (SI/SO) Registers, Serial In/Parallel Out (SI/PO) Registers and Parallel In/Serial Out (PI/SO) Registers.[11] Of these, the SI/SO is the simplest, consisting of a linear array of CCD cells. Depending upon the number of cells and the clock rate, various delays may be achieved. This structure is fundamental to CCD memories.

The SI/PO block has nondestructive taps along the delay line. By providing weights on these taps and using current summing, a device such as shown in Fig. 1 is obtained. If the tap weights are made equal to the components of a vector $h = (h_1, h_2, \dots, h_d)^T$ and a vector $u = (u_1, u_2, \dots, u_d)^T$ is loaded into the shift register, then the circuit will compute the inner product

$$y = h \cdot u \quad (1)$$

Matched filtering can also be accomplished by setting the tap weights equal to samples of the time inverse of the signal to be detected. At the n th clock time the output is then the weighted sum of the previous samples

$$y(n) = \sum_{i=0}^p h_i u_{n-i}, \quad n \geq p \quad (2)$$

The fixed tap weight structure in Fig. 1 is only a little more complicated than the basic SI/PO building block. Because of this and the fact that it implements a transversal or nonrecursive filter, it too is likely to become a basic building block in the near future.[12]

Some work has been done on structures similar to that in Fig. 1 in which the tap weights are variable [13, 14]. This would allow convolution or correlation of two arbitrary waveforms in addition to finding applications in time varying and adaptive control systems. To date, successful fabrication of

variable tap weight transversal filters has been somewhat limited compared to those in which the tap weights are fixed and represent a considerable challenge [15]. Because of their significant potential in control systems, structures will be suggested to demonstrate the need for continuing the development of these devices.

The PI/SO block is, of course, similar to the SI/PO block, except for inter-changing the input and output. The register is loaded by synchronously sampling all of its inputs. This configuration is shown in later figures.

At this point it may be well to note the conditions under which CCD processors might become practical for use in control systems. Classically, the control engineer would implement a transfer function with passive components and an amplifier. One condition that would make the CCD system preferable to the classical approach for a linear time invariant system will exist when the fixed tap weight transversal filter of Fig. 1 costs less than the controllers presently being used. Other conditions include the potential for requiring less space, power, volume and increased reliability. Timing functions not shown in the diagram increase the costs of the CCD controller. A bigger payoff is available in more complicated systems which require parameter changes not easily implemented with classical methods. In this case, the CCD controller competes with a form of computer.

Control Systems Examples

Linear Time Invariant Systems

There are several approaches for modeling automatic control systems which are suitable for CCD implementation. In the following, discrete convolution and difference equation models will be considered, although state space models can be used, also.

As a practical example, the following weighting function (minus an air-frame component) for a commercial aircraft will be considered. For the pitch rate input, it is given by

$$h(t) = 0.45te^{-0.62t} + 0.52e^{-0.62t} - 5.35e^{-5.0t} + e^{-4.2t}(4.8\cos 4.3t + 6.9\sin 4.3t) \quad (3)$$

The output $y(t)$ may be expressed as

$$y(t) = \int_0^t h(\tau)u(t - \tau)d\tau \quad (4)$$

which has the discrete form given in (2). For this case, the values of h to use in Fig. 1 for $d = 200$ can be obtained from (3) over a 10-second time interval. It would be practical to use even more values of h . For example, a 500-point CCD Fourier transform processor is being developed [16], and a 500-point CCD power density spectrum analyzer has been tested [17].

A typical operation cycle of the CCD processor might be 6 microseconds (i.e., a new value of $y(n)$ is available every $T = 6$ microseconds) where 1 microsecond is required to shift another sample of $u(t)$ into the shift register and 5 microseconds is required for the summation and operation of the zero order hold device. This is much faster than could be obtained with an ordinary digital computer.

An alternative way to realize this controller would be to discretize the

transfer function using Tustin's approximation, for example, and then convert to a difference equation formulation. In general, the result would be a recursive difference equation involving both the input and past values of the output. A recursive implementation is proposed in Fig. 2. It has been suggested[12] that CCD recursive filters may not be as cost effective as transversal filters because of the need for additional amplifiers on the chips. Consequently, the following example makes use of external feedback paths around non-recursive filters.

The general n^{th} order linear time-invariant difference equation is given by

$$y(k) = b_0 u(k) + b_1 u(k-1) + \dots + b_p u(k-p) - a_1 y(k-1) - a_2 y(k-2) - \dots - a_n y(k-n), \quad k = 1, 2, \dots \quad (6)$$

where $y(k)$ is the output at time $t = kT$ and $y(k-1)$, $y(k-2)$ are the output values previously stored in the top shift register of Fig. 2. Similarly, $u(k)$ is the input at $t = kT$ and $u(k-1)$, $u(k-2)$, \dots are the input values previously stored in the bottom shift register, and the coefficients a_1 and b_1 are the tap weights shown. Neglecting initial conditions, operation is begun by clearing both registers. Thereafter, values of $y(kT)$ are generated at each clock time.

An anticipated sampling interval for a system like this might be on the order of 10 microseconds where 6 microseconds are required for one cycle of operation of the two transversal filters and 4 microseconds are required for the final summation that provides $y(kT)$. Such a small interval in most systems would make it difficult to detect any difference from a continuous time. It is important to note that the minimum sampling rate is independent of the order of the equation, which is opposite the case where digital computers are used.

It seems reasonable to assume that this CCD implementation will be cost effective in the near future only for controllers of high order. In cases where filtering (e.g., Kalman filtering) is part of the controller, the order of the system will determine the equation order to be implemented. With CCD implementation it would appear that there is no need to obtain low order representations.

Systems in which Parameters Vary

In nonlinear systems or time varying systems, it is necessary to vary the tap weights as indicated by Fig. 3. Since the CCD controller cycle time is so much smaller than the response time of typical control systems, it seems reasonable to treat parameter varying problems as slowly varying linear systems. Thus, a microcomputer or other computer would calculate the tap weights at a relatively slow rate for the CCD controller.

Conclusions

The possibility of applying CCD technology to system controllers and filters has been suggested. This technology would seem to be most applicable to large systems or to applications requiring faster operation than digital computers can provide. Although the components required to implement the structures shown are not readily available at this time, it is anticipated that they soon will be.

Bibliography

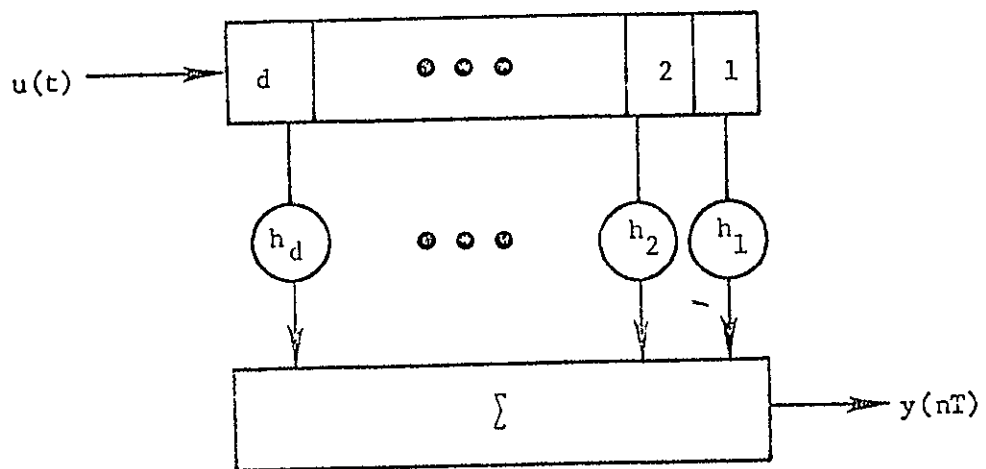
1. Electronics, p. 112, May 11, 1970.
2. Boyle, W. J. and Smith, G. E., "Charge-Coupled Semiconductor Devices," Bell Syst. Tech. J., Vol. 49, pp. 487-493, 1970.
3. Amilio, G. F., et al., "Experimental Verification of the Charge-Coupled Concept," Bell Syst. Tech. J., Vol. 49, p. 593, 1970.
4. Altman, L., "Charge-Coupled Devices Move in on Memories and Analog Signal Processing," Electronics, pp. 91-101, August 8, 1974.
5. Kosonocky, W. F., "Charge-Coupled Devices - An Overview," 1974 WILCON Professional Program, Session 2, Los Angeles, California, September 1974.
6. Butler, W. J., et al., "Practical Considerations for Analog Operation of Bucket Brigade," IEEE J. Solid State Circuits, SC-6, pp. 391-395, December 1971.
7. Buss, D. D., et al., "Charge Transfer Device Transversal Filters for Communication Systems," submitted to IEEE Trans. on Comm.
8. Boyle, W. J. and Smith, G. E., "Charge-Coupled Devices - A New Approach to MIS Device Structures," IEEE Spectrum, pp. 18-27, July 1971.
9. White, M. H. and Webb, W. R., "Study of the Use of Charge-Coupled Devices in Analog Signal Processing Systems," Final Report, Contract No. W00014-74-C-0069, Westinghouse Defense and Electronic System Center, May 1974.
10. New Products, Electronics, April 17, 1975.
11. Lampe, D. R., et al., "CCD's for Discrete Analog Signal Processing (DASP)," IEEE Intercon, March 1974, New York.
12. Buss, D. D., Bailey, W. H., Tasch, A. F., Jr., "Signal Processing Applications of Charge-Coupled Devices," Proc. Intl. Conf. on the Technology and Applications of CCD's, Edinburgh, September 25-27, 1974.
13. White, M. H., et al., "CCD MNOS Devices for Programmable Analog Signal Processing and Digital Memory," 1973 Intl. Electron Devices Meeting, Tech. Dig., pp. 130-133.
14. Tremann, J. J., et al., "Intracell Charge-Transfer Structures for Signal Processing," IEEE Trans. Elec. Dev., ED-21, pp. 300-308, May 1974.
15. Hartman, C. S., et al., "Programmable Transversal Filters Using Surface Wave Devices, Charge Transfer Devices, and Conventional Digital Approaches," Proc. Intl. Spec. Seminar on "Component Performance and Systems Applications of Acoustic Surface Wave Devices", Aviemore, Scotland, pp. 102-114, September 1973.
16. Benz, H. F., private communications, NASA, Langley, Va., May 1975.
17. Brodeisen, R. W., et al., "A 500-Point Fourier Transform Using Charge-Coupled Devices," Proc. of the Intl. Solid State Circuits Conf., February 1975.

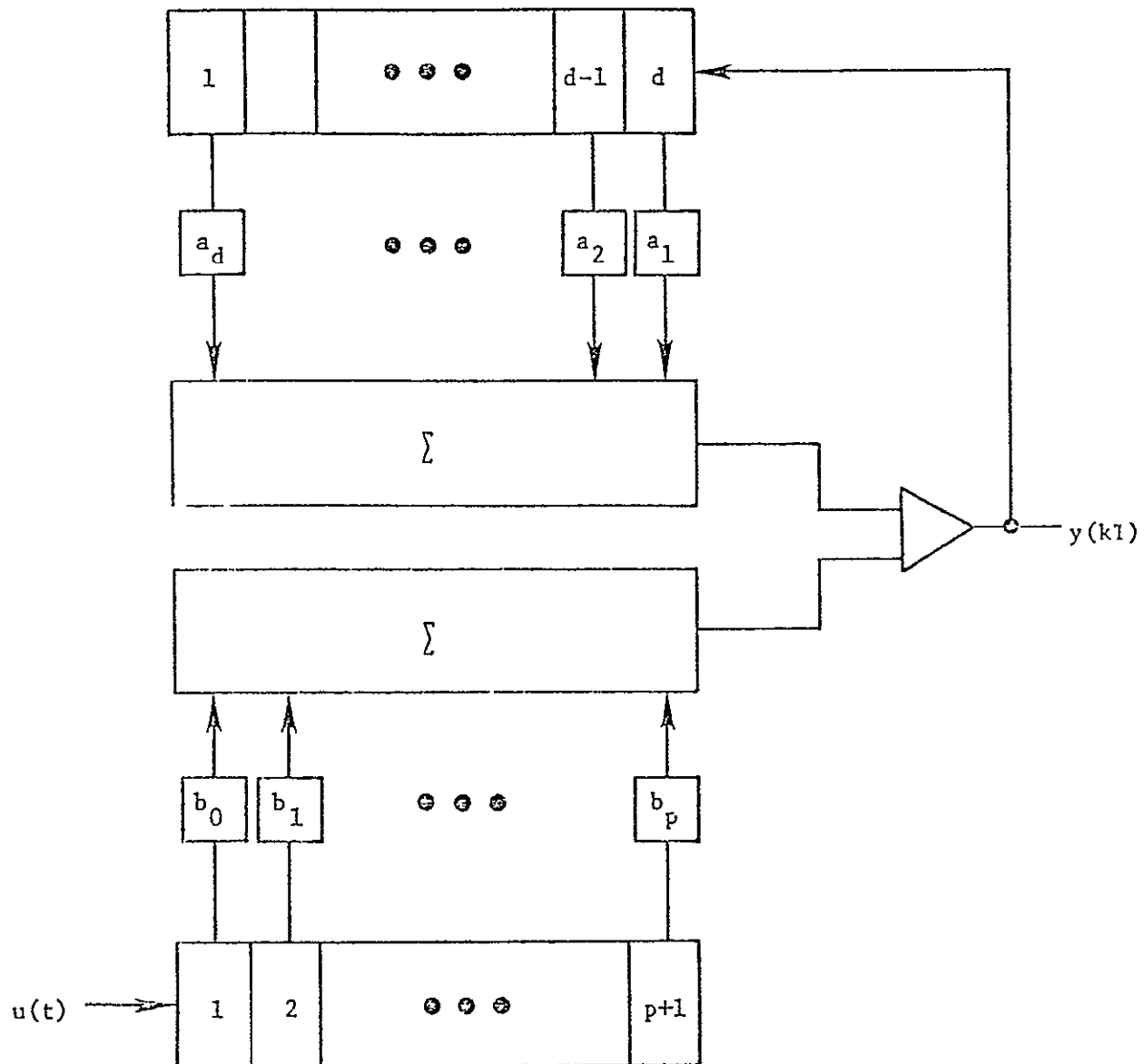
ORIGINAL PAGE IS
OF POOR QUALITY

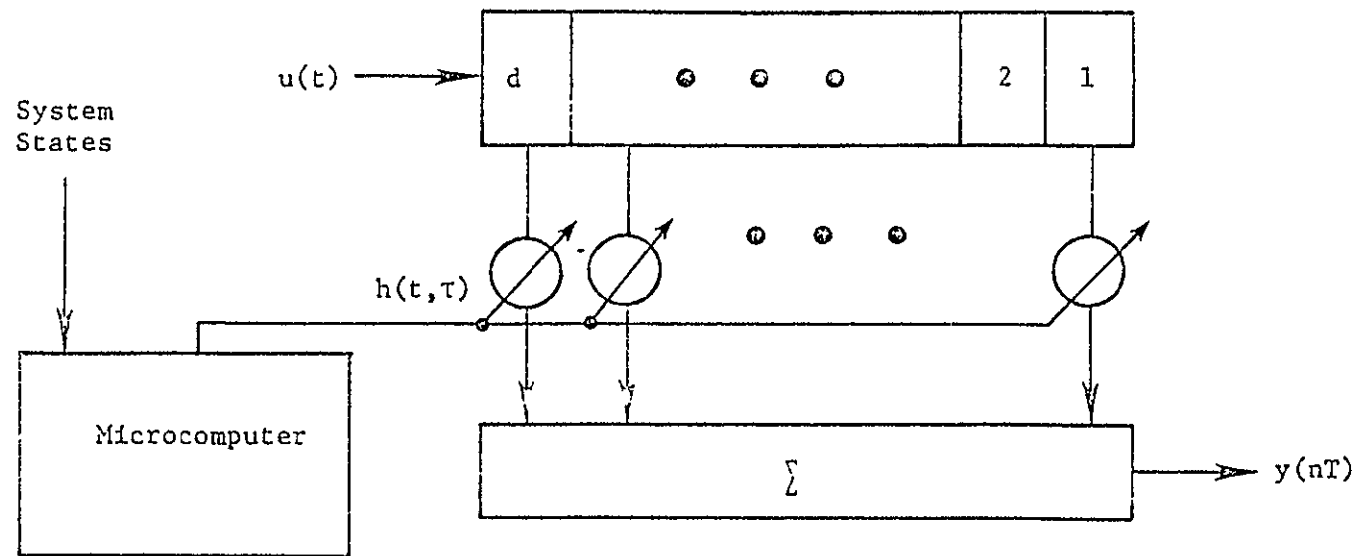
List of Captions

- Fig. 1 Linear Fixed Parameter System Using Discrete Convolution
Fig 2 Implementation of a Recursive Difference Equation
Fig 3 Linear Time Variable Parameter System Using Discrete Convolution
 with Variable Tap Weights

ORIGINAL PAGE IS
OF POOR QUALITY







DISTRIBUTION LIST

Copy No.

1 - 2	NASA Scientific & Technical Information Facility P. O. Box 8757 Baltimore/Washington International Airport Maryland 21240
3 - 5	National Aeronautics and Space Administration Langley Research Center Hampton, Virginia 23665 Attn: Mr. Charles Husson NASA Technical Officer
6 - 7	E. A. Parrish
8 - 9	E. S. McVey
10	J. N. Warfield
11	I. A. Fischer Office of Sponsored Programs
12 - 13	E. H. Pancake Sci/Tech Information Center Clark Hall
14	RLES Files

UNIVERSITY OF VIRGINIA

School of Engineering and Applied Science

The University of Virginia's School of Engineering and Applied Science has an undergraduate enrollment of approximately 1,000 students with a graduate enrollment of 350. There are approximately 120 faculty members, a majority of whom conduct research in addition to teaching.

Research is an integral part of the educational program and interests parallel academic specialties. These range from the classical engineering departments of Chemical, Civil, Electrical, and Mechanical to departments of Biomedical Engineering, Engineering Science and Systems, Materials Science, Nuclear Engineering, and Applied Mathematics and Computer Science. In addition to these departments, there are interdepartmental groups in the areas of Automatic Controls and Applied Mechanics. All departments offer the doctorate; the Biomedical and Materials Science Departments grant only graduate degrees.

The School of Engineering and Applied Science is an integral part of the University (approximately 1,400 full-time faculty with a total enrollment of about 14,000 full-time students), which also has professional schools of Architecture, Law, Medicine, Commerce, and Business Administration. In addition, the College of Arts and Sciences houses departments of Mathematics, Physics, Chemistry and others relevant to the engineering research program. This University community provides opportunities for interdisciplinary work in pursuit of the basic goals of education, research, and public service.